Automatic detection of fluorescein tear breakup sequence

A thesis submitted for the degree
of Doctor of Philosophy of
the Australian National University

Tamir Yedidya

THE AUSTRALIAN NATIONAL UNIVERSITY

18 September 2010


THE AUSTRALIAN
LIBRARY
NATIONAL UNIVERSITY

This thesis is submitted to the Department of Information Engineering, Research School of Information Sciences and Engineering, The Australian National University, in fulfilment of the requirements for the degree of Doctor of Philosophy.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university. To the best of the author's knowledge and belief, it contains no material previously published or written by another person, except where due reference is made in the text.

Tamir Yedidya

Tamir Yedidya

18 September 2010

Supervisory Panel:

Prof. Richard Hartley	The Australian National University and NICTA
Prof. Yogesan Kanagasingam	Lions Eye Institute and University of Western Australia
Dr. Nick Barnes	The Australian National University and NICTA

This thesis is based on materials from the following papers and abstracts:

Tamir Yedidya, Richard Hartley, Jean-Pierre Guillon and Yogesan Kanagasingam, "Automatic dry eye detection," MICCAI07, 2007, pp. 792 - 799 (**related to chapter 5**)

Tamir Yedidya, Richard Hartley and Jean-Pierre Guillon, "Automatic detection of pre-ocular tear film break-up sequence in dry eyes," DICTA08, 2008, pp. 442 - 448 (**related to chapter 5**)

Tamir Yedidya, Peter Carr, Richard Hartley and Jean-Pierre Guillon, "Enforcing monotonic temporal evolution in dry eye images," MICCAI09, 2009, pp. 976-984 (**related to chapter 6**)

Tamir Yedidya, Richard Hartley, Jean-Pierre Guillon. Automatic Fluorescein Break Up Time Detection in Dry Eyes. Abstract. Oral Talk in the Australasian Ophthalmic & Visual Sciences Meeting (2008) (**related to chapter 5 and parts of chapter 7**)

Tamir Yedidya, Richard Hartley, Jean-Pierre Guillon and Yogesan Kanagasingam, "Detection of the Tear Meniscus Shape Using Asyemmtric Graph-Cuts," ISBI, 2010, pp. 944-947 (**related to chapter 8**)

Tamir Yedidya and Richard Hartley, "Tracking of blood vessels in retinal images using kalman filter," DICTA08, 2008, pp. 52 - 58 (**related to Appendix A.**)

Acknowledgements

I would like to sincerely thank Prof. Richard Hartley for supervising me during my PhD in the Australian National University and NICTA. I always found his ideas and comments very insightful and without his help I would not have been writing this paragraph now. I also appreciate the research related freedom he gave me during my studies and the opportunity to visit other international labs. I would like to thank Prof. Yogesan Kanagasingam (yogi), who supervised, hosted and directed me during my four visits to the Lions Eye Institute in Perth. I would like to thank Dr. Jean-Pierre Guillon who constantly provided me with brilliant ideas regarding researching the tear film and helped me shape my thesis. Jean-Pierre has also let me do part of the clinical tests in his clinic, and I am thankful for that. I would like to thank Dr. Nick Barnes for his supervision and his will to always offer help in research and in solving urgent research issues.

I would like to thank my friends from RSISE and NICTA: Peter Carr, who was always happy to offer his help, Chris McCarthy whom I successfully shared a cubic for two years, Babak Rasolzadeh who always cheered me up (and also hosted me in Stockholm lab visit), Gary Overett (for accurate weather reports), Nathan Brewer (for candies), Tim Raupach (for the parties), Luping Zhou, Ramtin Shams, Sarah Hickmott, Jae-Hak Kim (who also hosted me in London lab visit), Christian Lingenfelser, Fangfang Lu, Dr. Hongdong Li, Dr. Jochen Trumpf, Manfred Doudar and the Big Guy. I would also like to thank the rest of the academic staff and students in RSISE, NICTA and the Lions Eye Institute.

Finally, I would like to thank my family and friends at home. Even though they were complaining that I am on the wrong side of earth, they always supported me in my decision to go to Australia to complete a PhD degree.

Abstract

Dry Eye Syndrome is a common disease in the western world, with effects from uncomfortable itchiness to permanent damage to the ocular surface. Almost 5 million Americans over 50 years old suffer from dry eye. A conservative estimate shows that approximately 17 million Americans have contact lens related dry eye - one of the main factors to contact lens discontinuation. In addition, the incidence of the disease is on the rise. Nevertheless, there is still no gold standard test that can reliably detect dry eye. One of the most commonly used tests by clinicians to detect dry eye is the Fluorescein Break Up Time (FBUT). However, results vary a lot between clinicians. Other tests such as observing the tear meniscus height are also performed regularly by the clinicians but not necessarily in conjunction with the FBUT test. Therefore there is a real need for a reliable, robust and operator-dependent method to evaluate dry eye. To our knowledge, no previous research has been conducted on automatic evaluation of dry eye in fluorescein images.

In this thesis, we present new algorithms to automatically detect various dryness signs and make a number of original contributions. The first problem we address is how to detect the dry areas in fluorescein videos of the anterior of the eye, which are captured using a portable camera. We present a new multi-step algorithm which first locates the iris in each image in the video, then aligns the images according to the location of the iris and finally analyzes the aligned video to find the regions of dryness. We produce a novel segmentation result called dryness image, which depicts the various degrees of tear film thinning over the corneal surface. Then, we demonstrate through experiments that there is a large variation in the estimated Break Up Time (BUT) between clinicians and no ground-truth can be defined. To overcome that, we define a new value based on the clinical definitions of the BUT. These definitions are converted to image processing properties and an estimate of the BUT is computed using temporal analysis of the aligned video. We demonstrate that our new value is

in the accepted range of the BUT values provided by the clinicians.

We present an extension to the dryness algorithm, which is based on transforming the video to a volume by considering each video frame as a slice in a 3D volume. On a volume, a temporal monotonic constraint can be applied between pixels in consecutive slices. The constraint enforces the clinical definition of tear film thinning over time - the amount of fluid cannot increase while not blinking. The constraint is applied directly into the cost function and the whole volume is segmented simultaneously using graph-cuts. As a consequence, the approach is more robust and less sensitive to alignment errors. Finally, we generalize the idea and explain how monotonic constraints can be applied to other imaging modalities.

In the last part of the thesis, we develop a new algorithm to evaluate the tear meniscus height and shape using graph-cuts. We formulate the segmentation problem using asymmetric cost functions and demonstrate its power and usefulness for the task. The asymmetry induces which directional moves are permitted in the minimization process and thus produces a result that adheres to the known shape properties of the tear meniscus. The iterative algorithm provides simultaneously the best segmentation result and shape prior of the meniscus.

Contents

Chapter 1

Introduction

Applying computer-vision techniques to medical images is not a new concept. Research involving MRI images of the brain has been going on for years now. Analyzing CT images of different body parts such as the kidney is also widespread. Even fundus images have received their share, mainly to detect lesions on the retina. Yet, there have been very little computer-vision research related to corneal images (images of the front of the eye) and the associated diseases, such as the Dry Eye Syndrome. There is no doubt that the gift of vision is invaluable. It is true that fundus images can (or at least theoretically) be used to detect lesions on the retina that might develop into severe diseases, such as diabetic retinopathy, which can cause blindness. It is also true that dry eyes will not cause blindness.

Are you one of 4.91 million Americans over 50 years old? Do you have repeated pain and irritative symptoms in your eyes? Do you have disturbed vision or blurry, foggy vision that only clears with a blink? Do you have problems doing common activities, such as reading, driving or using a computer? Is it inconvenient to instill lubricant eye drops so often that it even affects your social and workplace interactions? Or maybe you cannot tolerate your contact lens any longer that you had to discontinue wearing them? These questions can appear on a questionnaire to diagnose dry eye. In our modern society, the quality of life is highly valued, and most of these questions relate to one or other aspects of quality of life. In some cases, the burden of dry eyes can be quite severe, limiting us in our everyday activities or even causing an eye infection.

Five million sufferers comprises a high percentage from the elderly population in America. It has also been found that the incidence of dry eye has increased by 57.4%

from 1991 to 1998. Therefore, the prevalence of dry eye is on the rise. Dry eye is a common yet frequently under-recognized clinical condition which keeps challenging researchers to understand its cause. This long list of dryness symptoms, the high number of sufferers, the chronic nature of the disease and the lack of reliable diagnostic tests to detect dry eye are among the reasons that the 200-pages International Dry Eye Workshop (DEWS) report were published in 2007 [100]. This report redefines dry eye syndrome, explains its causes, symptoms and treatments and discusses methods to diagnose dryness and other related clinical material. The report is mainly aimed at medical researchers and clinicians and strongly conveys the need for further research in the field.

1.1 The challenge of diagnosing dry eye

A clinician regularly performs a diagnosis test to diagnose dry eye and it is a relatively simple task. According to [68], the most commonly used method is the Fluorescein Break Up Time (FBUT). The idea is very simple: a small amount of fluorescein is instilled in the eye. The eye is viewed using a slit-lamp (where the slit is covered) with a blue light and a yellow filter. The light excites the fluorescein and dark spots will start appearing on the cornea if the eyes are kept open and dryness exists. The clinician uses a stopwatch and the time taken for dark spots to appear on the cornea after a blink is called Break Up Time (BUT). This is the value returned from the test and a cutoff point (in seconds) is defined to diagnose dry eye.

Nevertheless, this test suffers from a number of limitations making it subjective and operator-dependent. In addition, a single cut-off point is probably a harsh measurement to analyze dryness and more symptoms (some are provided by the FBUT test but not used) should be considered. This point of view is also clearly expressed in the DEWS report. The committee mentions that no single diagnostic test can be performed to reliably distinguish individuals with or without dry eye. Also there is little correlation between patients' symptoms and the results of selected clinical tests. This is explained by the lack of reported repeatability when using the same test at different times.

The committee concludes that no "gold standard" exists for the diagnosis of dry eye. New diagnostic tests should be developed to better facilitate clinical research and the understanding of the factors affecting dry eye. Therefore, there are challenging tasks to define more robust measurements to estimate dryness which are repeatable, not subjective and not operator-dependent. Such measurements could be part of a system which is capable of simultaneously producing results based on a few diagnostic criteria.

However, to our knowledge, no research has been conducted to built a system that will automatically detect dry eye. Such automatic system that is based on the FBUT test should be able to find the BUT (or a clinically related value) in a repeatable way. It should also provide a finer analysis of the tear film break-up pattern, such as shape, location and size [10] and one or more diagnostic results that can point towards the origin of the dryness. Hence, if we develop such a system, it can be installed in the clinician's clinic to ease the detection of the disease. It will also be a useful tool to collect mass data to be recorded and used later to understand better the cause of dryness. Finally, it solves some of the important issues described in the DEWS report, such as repeatability, no need to define a single threshold and serves as the first step towards an accepted diagnostic criteria of dry eye for epidemiological studies.

1.2 Problem definition

In this thesis, we investigate how to automate the diagnosis of dry eye. We capture image sequences (videos) using a hand-held portable camera. Therefore, we tackle the dryness problem using a similar input to what the clinician sees when performing the FBUT test, but maximize the information that is retrieved. Below we list specific issues this thesis addresses for the automatic analysis of dryness:

Detection of the iris and eyelids in the video: The dry areas can appear only over the iris, therefore the exact location of the iris has to be found. Our detection task differs from traditional iris fitting algorithms, since our eye images are recorded after instilling fluorescein. We show how the eyelids are detected using a polynomial fitting method and the iris is found accurately using circle detection.

Aligning the iris in the image sequence: In order to make our solution more accessible, we have used a portable video camera which can be carried outside the clinic (of course, the regular slit lamp can be used as well). As a result, the location of the iris throughout the video can not only change due to the patient's change of gaze but also due to camera movements. Given the estimated location of the iris in each image, we use Lucas-Kanade alignment method and Levenberg-Marquardt optimization algorithm to align the iris over the video. Thus, fixating the location and size of the iris. This step is crucial for any further dryness analysis.

Evaluating dry areas and the BUT: In general, the clinician only measures the BUT when performing the FBUT test. Therefore, we investigate the repeatability of the BUT value among clinicians and the reasons for the differences. We define a procedure to robustly compute a similar clinical measurement to the BUT, which is based on temporal image properties. Taking advantage of the alignment result, a wide range of dryness related symptoms is evaluated and reported.

Segmentation of the tear meniscus: As mentioned before, the FBUT test is not always enough on its own to detect dry eye. Using static images from the video and the detected location of the eyelids, we estimate the height and shape of the tear meniscus. We use a graph-cut optimization algorithm which takes advantage of the known shape properties of the tear meniscus.

1.3 Contributions

The thesis contributes the following:

- We show how to compute the Break Up Time in fluorescein images in an automatic way without any input from the clinician. We define a new clinical value called Digital Electronic Break Up Time (DEBUT), which is robust and invariant to eye movements, illumination changes and operator. We tested the DEBUT value on over 100 videos.

- We detect and analyze in an automatic way a variety of symptoms related to dryness, such as: individual break areas and their location, progress, shape and

size; the black line that appears near the eyelids and is related to lid originated dryness; global iris information regarding the progress of the break. To our knowledge, these symptoms have not been analyzed in an automatic way before. Thus there is a very limited record of these symptoms and they are not available for further research.

- We carried out a set of experimental tests to find intra-observer and inter-observer differences when estimating the Break Up Time (BUT). Our experiments show meaningful inter-observer variance (between clinicians), thus stressing the lack of repeatability of the BUT value when computed by clinicians. This finding explains why a more robust method and value are desirable and corroborates the need in introducing the new DEBUT value.

- We present a graph-cut based method which transforms a (dry-eye) video of 2D images to a 3D structure and enforces a monotonic spatio-temporal constraint for a better segmentation. The method increases the reliability of the detection of dryness related symptoms. In addition, it offers a general approach on how to add monotonic (temporal) constraints to volumes, even if by nature they are only 2D and suggests a solution based on graph-cuts.

- We present a new method to segment and analyze the tear meniscus height, shape and irregularity. We show how asymmetric cost functions are used in a graph-cut based algorithm to enforce directional constraints. These constraints exploit prior knowledge about the desired shape of the object and enforce the segmentation result to be of that shape.

1.4 Overview

In **chapter 2**, we provide a general overview of medical background related to the tear film such as: how tears are produced, what is the dry eye syndrome and various clinical tests to assess the tear stability. We also focus on the Fluorescein Break Up Time test, which is the test we have used throughout the thesis to assess dryness. **Chapter 3** is an overview of related literature to our dryness detection algorithm. We discuss

other automatic methods for iris detection, alignment methods and non-automatic methods for BUT detection. More related work appears in the introduction section of specific chapters. **Chapter 4** gives an overview to graph-cuts and Markov Random Fields, which are used in the thesis in both chapters 6 and 8.

The main work of this thesis is presented in chapters 5 to 8. In **chapter 5**, our main algorithm to analyze dryness is given. The chapter is divided into three main sections: detection of the iris and eyelids in all video images, aligning the images in the video and analyzing the video for dryness symptoms. We also present the new concept of Digital Electronic Break Up Time and show how it is computed. **Chapter 6** is an extension of chapter 5, and it shows how graph-cuts and monotonic constraints can aide towards improved detection of the dry regions and the DEBUT value. It also includes a general algorithm demonstrating how to enforce monotonic constraints in graph-cut based solutions. **Chapter 7** describes the experiments we have carried out. We show the correctness of our algorithm and the DEBUT value when compared to the clinician's BUT. In addition, through the experiments we demonstrate a major problem in the existing FBUT test: large inter-observer variance among clinicians. In **chapter 8**, we reveal our graph-cut algorithm to assess the tear meniscus, and we show how an asymmetric cost function can be used to enforce directional and shape constraints. In **chapter 9**, conclusions and discussions are presented.

In addition, we include in **Appendix A** our method to detect blood vessels in retinal images. Due to the difference in image modality from the rest of the work (corneal images), we add this material as an appendix. In this chapter, we describe our tracking algorithm that uses the Kalman filter to segment the blood vessels.

Chapter 2

Medical Background

This chapter provides medical background related to the tear film and dry eyes. Blinking, which spreads the tear film over the cornea, is a simple action we perform thousands of times every day. However, the blinking process is actually not that simple. See Fig. 2.1 about our blinking habits. Unfortunately, this whole process can be easily interrupted and may cause discomfort related to dry eyes. Interestingly enough, dry eyes are very prevalent in the western world, however, there is no single objective technique to evaluate dryness. We will discuss these issues in this chapter. The chapter is organized as follows:

- Tear film structure - the different layers.
- Dry eyes - definitions, production of tears, symptoms and treatment.
- Techniques to evaluate dryness, the fluorescein break up time test, our clinical method and meniscus evaluation

2.1 The tear film

The tear film is a highly specialized and carefully structured moist film which covers the conjunctiva and the cornea. Abnormalities of the film can rapidly result in a serious dysfunction of the eyelids, the conjunctiva and the cornea. The presence of a healthy tear film is important as it provides a smooth regular optic surface covering the corneal epithelium (a layer of cells). It also adheres to the conjunctival surfaces and keeps them moist and lubricated. As tears flow across the ocular surface, they

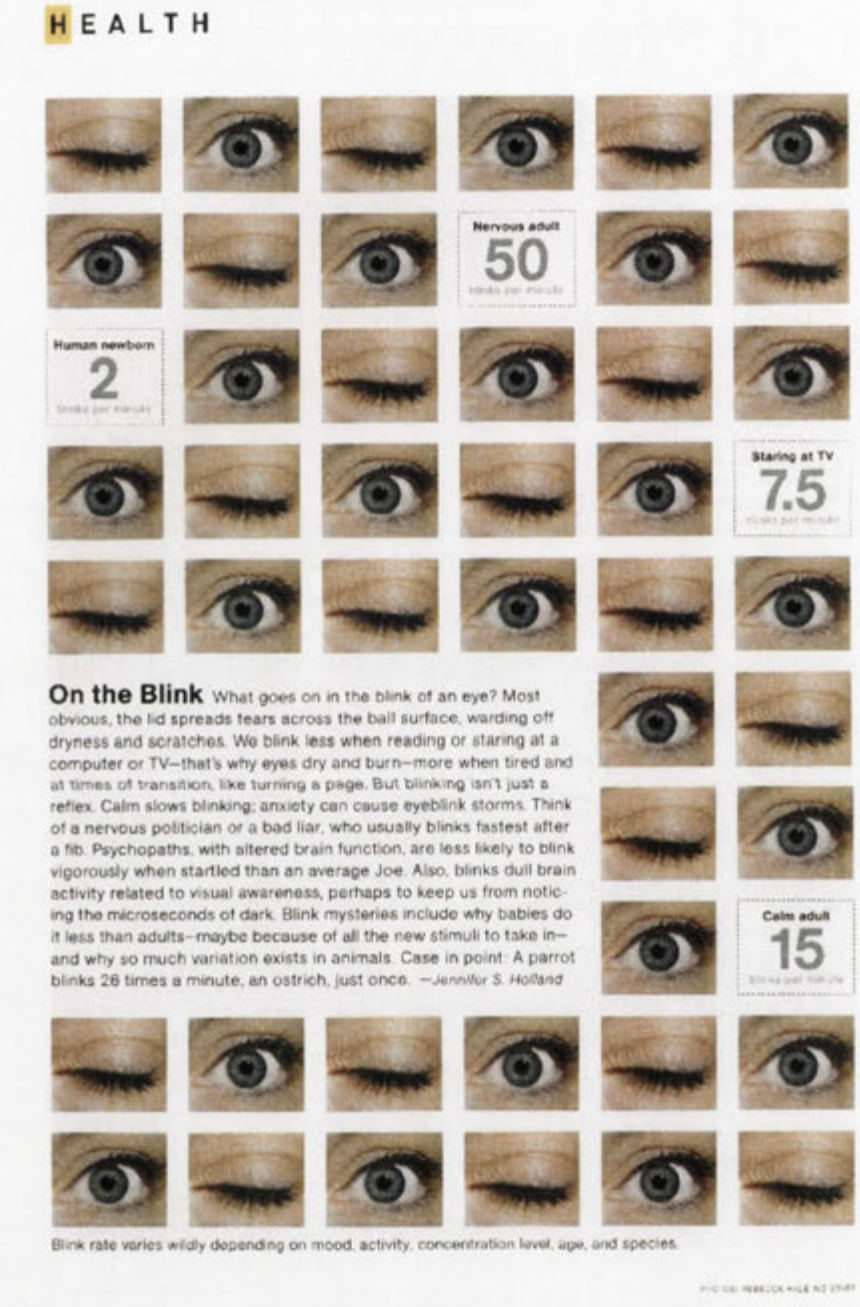


Figure 2.1: *Courtesy of National Geographic Magazine. Blinking. Everyone is blinking, but how often?*

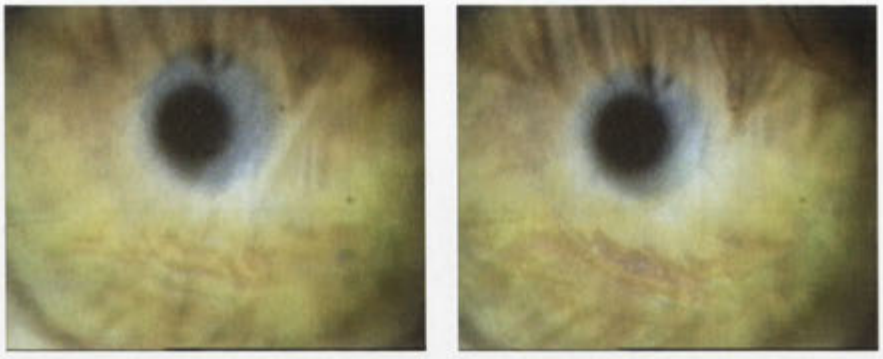


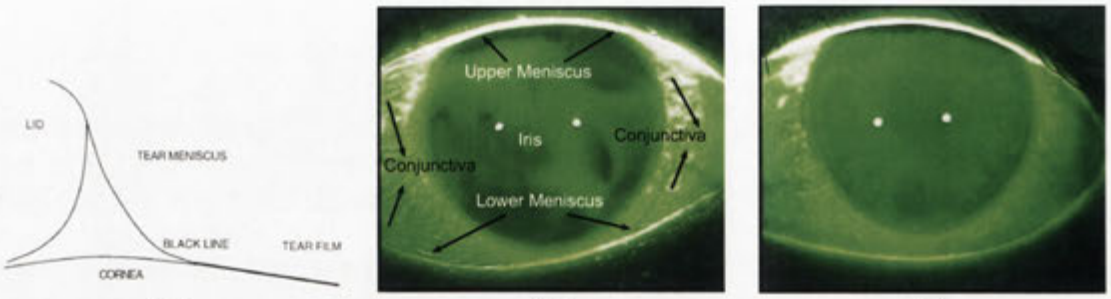
Figure 2.2: *The tear film highlighted. Two images taken 0.5s and 1.5s after the blink respectively. The tear film is the colored wavy stripes that can be seen in the bottom of the image. Of course, it covers the whole of the cornea, but cannot be seen clearly.*

flush debris and foreign matter. Finally, the tear film is the first line of defense against ocular surface infection. In Fig. 2.2 the tears (tear film) can be observed at the bottom part of the iris. Usually it is not an easy task to see them that clearly as it depends on many factors related to the structure of the tear film.

The design of contact lens placed before the cornea is also highly related to understanding the tear film structure. Contact lens is used to correct refractive error and is bathed in the tear fluid. Dry eye problems are responsible for a high proportion of contact lens failures. Therefore, it is important to understand their effect on the tear film.

2.1.1 Tear volume

The total volume found within the palpebral aperture (eyelids) has been estimated to be between $7\mu l$ to $10\mu l$ and up to 90% of the tear volume is found in the superior and inferior marginal tear strip, also known as tear meniscus and tear reservoir , see Fig. 2.3(b). The remaining volume is spread over the ocular surface to form the very thin pre-ocular tear film (POTF) . When blinking is permitted, it appears that for a normal palpebral, any amount above $10\mu l$ will be flushed out, however, it varies with the size of the palpebral. In Fig. 2.3(a) a schematic model of the tear meniscus is depicted, showing the contact points between the meniscus and the cornea and the tear film. The almost triangular shape formed between the meniscus and the lid and the cornea



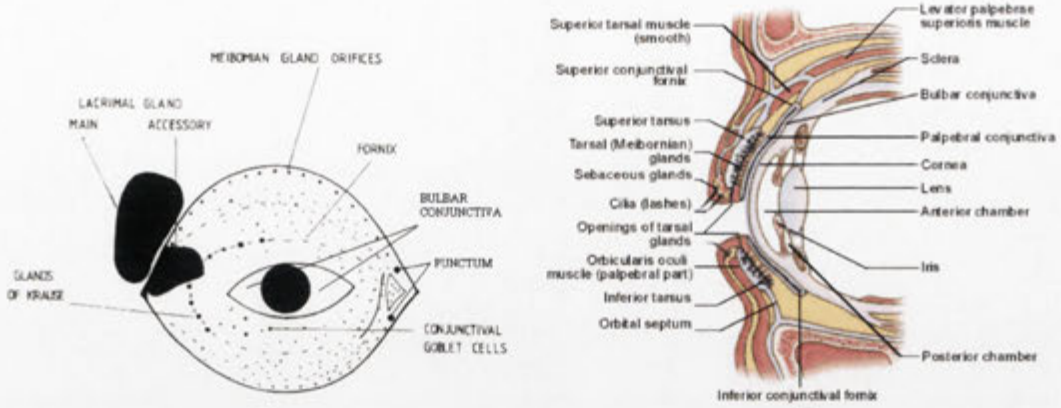
(a) (b) (c)

Figure 2.3: *Explaining the tear meniscus. (a) Schematic diagram of the tear film boundaries with the tear meniscus and the cornea. Courtesy of F.J. Holly, 1978 [47] (b) Basic eye terms depicted when using fluorescein, as this is the modality we will use in this work. (c) A thin black line is visible next to the upper eyelids*

is where the fluid is. Fig. 2.3(b) explains the basic terms using an image taken after instilling fluorescein. As we will mainly use fluorescein imaging in our thesis, we will use mostly this kind of imaging modality for explaining the eye-related background.

The average total thickness of the POTF, indicates a volume of approximately $1\mu l$ for the POTF, which corresponds to only a very limited amount of the total tear volume. As most of the fluid is found in the tear meniscus, using fluorescein staining (see section 2.3.2) reveals a black line extending along the corneal surface just within the lid margins. These black lines separate the POTF from the tear menisci and represent an area of thinning of the tear film [86]. That junction zone of minimal thickness corresponds to the area where the tear film is at its greatest instability. As a result, the surface tension forces make the formation of a continuous film nearly impossible. The black line is visible in Fig. 2.3(c) near the upper eyelids appearing darker than the colored iris. The detection of the black line is clinically important as it can be related to reservoir related dryness. Another reason the black line is created can be attributed to improper blinking. In that case the tear film is not spread evenly over the iris.

The formation of the tear meniscus depends on the balance between the negative pressure induced by its concave surface and the hydrostatic pressure due to the height of the fluid. In a fluorescein image, such as Fig. 2.3(b), the meniscus height is visible throughout the cornea and the conjunctiva. The meniscus height becomes shallower towards the left hand side. As in the ocular environment, the amount of fluid is lim-



(a) (b)

Figure 2.4: *The eye structure. (a) A frontal view of the eye showing the origin of tear film components. The iris in drawn in the middle and the surrounding glands are depicted. Courtesy of Tiffany and Bron, 1978 [106] (b) A sagittal view of the eyelids and anterior orbital structures. The figure provides an idea of the different layers of the eye. The lens, iris and the cornea are at frontal layers. Most of the other components are depicted as well, but are not necessary to understand the thesis. Courtesy of [122]*

ited and an imbalance is created because the tear film cannot replenish the fluid fast enough. The aspiration of fluid (removal of fluid) is the result of an unsaturated or thirsty meniscus and leads to localized thinning of the film. Observing the black line and the curvature of the meniscus gives information as to the degree to which it is unsaturated. The smaller the radius of the curvature, the more thirsty the meniscus is. When the meniscus gets larger and less curved, it becomes less thirsty as a large volume of fluid is available.

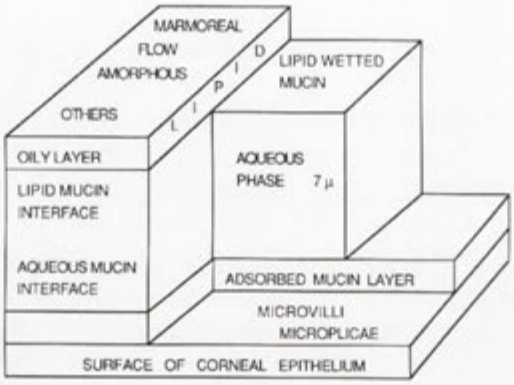
2.1.2 Tear film structure

The present thinking on the structure of the tear film is well established after Holly and Lemp [49] and stems from the basic description of a tri-laminar structure.

I will now describe briefly each of the layers in the tear film as depicted in Fig. 2.5. The interested reader can find a more elaborated description in [68].

Corneal epithelium

The superficial part of the cornea is the stratified (layered) epithelium which consists of zones of nucleated (having a nucleus) cells. The different cell layers represent dif-



(a)

Figure 2.5: *The pre-corneal tear film structure. Above the corneal epithelium (cells) are a mucous layer, a watery (aqueous) layer and a lipid layer. The aqueous layer is the thickest layer at about 7μm, the lipid layer is 0.1μm and the mucus layer is less than 0.05μm. Courtesy of J. P. Guillon, 1987 [37]*

ferent stages of cell development. As the cells are pushed towards the surface, they are continuously replaced by new cells. In severe cases of dry eyes, these cells are damaged, which results in damage to the ocular surface.

Mucous layer

The thickness of the mucous layer remains under debate. The periacinal source of mucous are the goblet cells of the conjunctiva. The POTF is dependent upon a constant supply of mucous, which is used to maintain the corneal and conjunctival surfaces in the proper state of hydration. Mucous is secreted by goblet cells, which are distributed over the conjunctival surface.

The ocular mucous performs several functions. Among the most important is the lubrication, allowing the eyelid margins and palpebral conjunctiva to slide smoothly over one another. Another important function is protection of the epithelial surface. Mucus threads are responsible for covering foreign bodies to protect the cornea and the conjunctiva from abrasion. The surface of the mucus is the first solid layer encountered by invading material such as bacteria. Therefore, the rapid self-repair of the mucus is essential in protecting the eye against drying effects and bacteria.

Aqueous layer

The aqueous phase of the tear film accounts for over 90% of its thickness and is made up of 98% water. The phase originates from the main lacrimal gland (see Fig. 2.4(a)) and the accessory lacrimal glands of the conjunctiva. The lacrimal gland is situated just above the outer corner of the eye behind the upper eyelid.

Lipid layer

The superficial lipid layer is a relatively thin, oily layer and is derived from three sources, all of which are in the eyelids. The majority of the lipid is produced by the meibomian glands which are embedded in the upper and lower eyelids. Blinking fills and releases meibomian gland fluid from the ducts and spreads it over the aqueous layer. The layer is important in retarding evaporation of the underlying aqueous layer. A four-fold increase in tear evaporation is observed when the human lipid layer is absent. It also prevents contamination of the tear film by skin lipids, which differ in composition from tear film lipids and can destabilize the lipid film.

Assessing the lipid layer has an important role when fitting contact lens. The principal alteration to the tear film structure upon insertion of contact lens is to the lipid layer. Inserting a contact lens within the aqueous layer creates a considerably thinner layer of fluid on which the lipid can lie. This results in a thin lipid layer or the absence of a lipid layer over rigid contact lens.

Similarly, the stability of the tear film is affected. For a stable film to be formed, the contact lens has to be entirely compatible with the tear fluid. Unfortunately, the hydrophobic nature of contact lens material, particularly rigid lens, often excludes the formation of such a film. Consequently, tear film stability is reduced in the presence of contact lens. Thick lipid layers are associated with increased tear film stability. A POTF lipid layer which is thick and stable is more likely to form a continuous lipid layer over the surface of a contact lens than a thinner one.

2.2 Dry Eye Syndrome

Dry Eye Syndrome (DES) [100] is a common disorder of the tear film, affecting a significant percentage of the population. It is caused by qualitative and quantitative abnormalities in the tear film layer. Tears are produced by the lacrimal gland (see Fig. 2.4(a)) and distributed over the ocular surface by blinking. Some of the tears evaporate and the rest are drained. Abnormalities in this process such as: decreased tear production, excessive tear evaporation, abnormality in the production of mucus, decreased blinking and not closing properly the eyelids, are the cause of dry eyes. In 1995, the NEI/Industry workshop broadly defined dry eye as

> a disorder of the tear due to tear deficiency to the interpalpebral ocular surface associated with symptoms of ocular discomfort. [73]

The requirement of symptoms is important as it is not included in the definitions established in all nations. In 2007, in the special International Dry Eye workshop (DEWS) [18], a subcommittee tried to refine this definition. The main challenges are that there is no single diagnostic test that can reliably distinguish between dry eye patient and a non dry eye patient (see more about dry eye tests in Sec. 2.3). There are numerous tests that are in use by clinicians to diagnose dry eye, but there is no consensus what combination of them should be used. Furthermore, there is lack of correlation between the symptoms and the results from the tests. As dry eye is a symptomatic disease, the symptom questionnaires are amongst the most repeatable. Therefore, it had to be addressed in the new definition. Finally, the new gold standard definition is:

> Dry eye is a multifactorial disease of the tears and ocular surface that results in symptoms of discomfort, visual disturbance, and tear film instability, with potential damage to the ocular surface. It is accompanied by increased osmolarity of the tear film and inflammation of the ocular surface.

The inflammation is caused by the lack of fluid, which triggers a reaction from our body to cope with the problem. The reaction can be the pumping of more blood into

the region, making the area look red. It is often accompanied by staining symptoms, which signify damage to the epithelial cells.

Based on data from the largest studies of dry eye to date, it has been estimated that about 3.23 million women and 1.68 million men over 50 years old in America have dry eyes. Tens of millions more have less severe symptoms. In Australia, in the "Blue Mountains" survey, out of the 1075 participants at least 50 years old, 16.6 percent had dry eye symptoms [23]. An overall summary suggests that the prevalence of dry eyes lies somewhere in the range of 5-30% in the population of 50 years and older. The variance might be related to the exact definition of dry eye used in each of the studies. Dry eyes are more prevalent in older age females. As a result, the incidence of dry eyes is expected to rise in the following years, since it is estimated that the number of elderly people will increase by 100% in the US in the period between 2000 and 2050.

2.2.1 Production and distribution of the tears

As dry eye is caused by abnormalities in the tear film layer, it is of interest to first understand how the tears are produced, secret and evaporate. I will briefly talk about the production of tears by the lacrimal gland (basic tearing), distribution by blinking, and evaporation from the ocular surface. Further material can be found at [68, 110].

Production of tears [110]

It has been suggested that the lacrimal gland steadily produces tears at a certain level without any stimulation [57]. This concept is called basic tearing and is related to tearing caused by blinking and tear evaporation. It is assumed that the tears come from the accessory lacrimal gland, whereas reflex tearing comes from the main lacrimal gland. The ocular surface is thus always covered by a tear film layer.

Reflex tearing results from irritation of the eye by foreign particles or from the presence of irritant substances. Reflex tears attempt to wash out irritants that may have come into contact with the eye. The tears are produced by strong physical or emotional stimulation to the lacrimal gland. Even if basic tearing is decreased, reflex tears can provide the ocular surface epithelium with substances necessary for proper

epithelial healing.

Tear distribution by blinking

Tears secreted into the upper fornix are moved towards the lacrimal puncta and form (by gravity) the lower tear meniscus. The tears are spread over the ocular surface and are drained through the lacrimal puncta by blinking. The blinking proceeds from temporal to nasal pushing the tears towards the punctum. The temporal side and the nasal side will be at opposite sides in left and right eye images. For example, in Fig. 2.3(c), the temporal side is on the right (where the eyelids merge) and the nasal side is on the left. Blinking abnormalities can cause improper tear distribution and hence lead to ocular surface problems.

The lids also play a role in the even distribution of the tears over the ocular surface. First of all, the eyelids pull the lipid layer with it as the eye opens. The oily layer then drags the aqueous layer upwards from the meniscus, as a result of differences in surface tension between the layers. Thus, the lipids help spreading the fluid over the corneal region and creating the tear film.

Human blinking averages 15-20 blinks per minute under relaxed conditions. While blinking is primarily a response from the ocular surface, other factors can play a role as well, such as environmental factors, reading or the use of computers. Blinking can also be incomplete, where the upper eyelid does not finish its movement downwards, in which case some portion of the cornea is always dry. In dry eyes, the blink rate is likely to be increased, in order to compensate for the decreased tear production and maximize the tear supply to the ocular surface. It has also been shown that contact lens wearers blink less and exhibit a higher proportion of incomplete blinks than non-wearers. This reduces the stability of the tear film and thus stresses the importance of performing blink exercises.

Tear evaporation

A small amount of tear fluid is lost passively by evaporation. About 10% of the total tear volume evaporates while the rest are drained through the punctum. In patients

with a small lipid layer, evaporation becomes more important. Instilling any drop onto the ocular surface disrupts the structure of the tear film and causes an increased rate of evaporation. Similarly, all contact lenses disrupt the superficial lipid layer and cause an increase in tear evaporation.

2.2.2 Morbidity of dry eyes

The symptoms of dry eye consist of:

- Dryness and grittiness feeling in the eye.
- Redness of the eyes
- Headaches and blurred vision
- Increased uncomfortableness when wearing contact lens for long periods.
- In severe cases, damage to the ocular surface.

The impact of dry eyes on the quality of life is quite high, as it can degrade the performance of common vision-related daily activities, such as driving. The cost of treatment, such as instilling eye drops, and the lack of a cure for dry eye add to the impact of this problem. A recent survey investigated to what extent dry eye patients are affected in their daily life. Patients with DES were significantly more likely to report problems with reading, using a computer, watching television and driving at night. Overall, DES patients were three times more likely to have difficulties than those without DES.

Dry eye patients can be divided into two categories: those with Sjogren's Syndrome (SS) and those without it. The ocular abnormalities are much severe in SS patients. Both groups lack tears, but the difference is in the ability of non-SS patients to produce reflex tears. SS syndrome is related to abnormalities in the immune system and is a severe form of dry eye. SS patients have a much increased feeling of fatigue and can also suffer from depression.

Except for symptoms of inconvenience, dry eye can lead to impact on the visual acuity. Visual complaints are highly prevalent among dry eye patients. It is usually

described as blurry vision that clears with a blink. A disrupted tear film in the central cornea can result in transient vision changes [111]. These changes can lead to a decrease in workplace productivity.

Finally, a contact lens (CL) placed before the cornea, to correct refractive error, is bathed in the tear fluid. Dry eye problems are responsible for a high proportion of contact lens failures, such as intolerance, reduced wearing time and discontinuation of contact lens wear. Therefore, it is important to understand the effect contact lens have on the tear film so that the most suitable materials will be chosen.

2.2.3 Risk factors

The risk factors for dry eyes are very diverse and include:

- female gender
- age
- diet
- climate
- contact lens wear
- the use of computers.

Vitamin A deficiency is a well-recognized risk factor for dry eye [102], as well as diet low in omega 3 [88]. Risk factors related to alcohol and smoking and caffeine have conflicting results. Indoor environment is also often associated with dryness symptoms such as burning and stinging. Ocular dryness due to increased tear evaporation may be caused by low humidity, air-conditioned rooms and increased blink rate. Computer use is also highly related to dry eyes, as the prolonged concentration on the screen is associated with decreased blink rate and is related to the "computer vision syndrome" [11]. Dry eyes appear to contribute a major component of symptoms to the syndrome.

A significant number of CL-wearing patients experience dryness, with 50-75% of wearers report symptoms of ocular irritation. This is equivalent to a more than 17

million Americans with CL-related dry eye [6]. The lenses tend to absorb the tears and proteins from the tear film. As a result, the rubbing of the lenses against the conjunctiva creates dry spots on the lens surface. Such symptoms are often reported as the cause of the discontinuation of wearing CL.

2.2.4 Treatment

In most cases alleviating the symptoms of DES can be achieved by using artificial tears, lubricants and gels, which provide more moisture to the surface of the eye. Most of the over the counter products are preservative-free and can be used as often as needed. It is difficult to prove that any ingredient in an ocular lubricant acts as an active agent. This is either because it is not possible to detect the effects in clinical trials or because the currently available agents do not have any discernable clinical activity beyond a lubrication effect.

The foremost objectives in caring for patients with dry eye disease are to improve the patients' ocular comfort and quality of life, and return the ocular surface and tear film to the normal homeostatic state. Although symptoms can rarely be eliminated, they can often be improved, leading to an improvement in the quality of life. It is more difficult to demonstrate that topical lubricants improve the ocular surface and the tear film abnormalities associated with dry eye. Until agents are developed that can restore the ocular surface and tear film to their normal homeostatic state, the symptoms and signs of dry eye disease will continue.

In more severe cases, minor surgical procedures may help. A procedure known as Punctual occlusion can help by decreasing the drainage of tears. Puncta (see Fig. 2.4(a)) is the opening near the corner of the eyelids where the beginning of normal tear drainage occurs. The procedure places small plugs inside these openings and keeps the tears from flowing down the drainage. These plugs work in the same way as stoppers put in a sink to keep the water from flowing down the drain. As a result the eyes stay moist even if there are a few tears.

Interestingly, contact lens may also help to protect and hydrate the corneal surface in severe dry eye cases. Several different contact lens materials have been evaluated.

Improved visual acuity and healing of corneal cells defects have been reported. However, there is a risk of corneal infection associated by the use of contact lens.

In addition, tear-stimulation methods, anti-inflammatory therapy and biological tear substitutes can be used for the therapy of dryness (see details in [93]). Some of these methods have only been approved in the last two decades, making the shift from the usage of lubricants to strategies that stimulate natural production of tears and maintain healthy ocular surface.

2.3 Assessing the tear stability

2.3.1 Break-Up of normal tear film

The pre-ocular tear film in humans does not remain stable for long periods of time [48]. When blinking is prevented, the tear film ruptures within 15 to 40 seconds exposing dry spots of uncovered epithelium over the cornea [75]. The tear film decreases in thickness by around 10 percent through evaporation between two consecutive blinks. In dry eyes, the tear film usually ruptures before the blink, thus exposing the cells. Sometimes, the tear film may not rupture, but the blink rate is increased to compensate for the inconvenience associated with the dryness. The time between the rupture of the tear film and the subsequent blink is also of importance: the longer the time is, the less comfortable the patient feels and more damage is done to the cornea. The tear film is formed and dragged into place once the eyelids open after a blink. The eyelids pull the lipid layer with it and then the lipid drags the aqueous layer upwards from the meniscus. Once formed, the property of keeping the integrity of the tear film is called tear film stability. The lipid seldom stays stable for long periods of time, as some of the lipids will migrate to the epithelium surface and contaminate the mucin layer. This creates a high tension between the layers and makes areas of the mucus layer hydrophobic. The tear film in these areas becomes unstable and thus leads to the formation of non-wettable areas of increasing size. This process is called Break Up of the tear film and is depicted in Fig. 2.6.

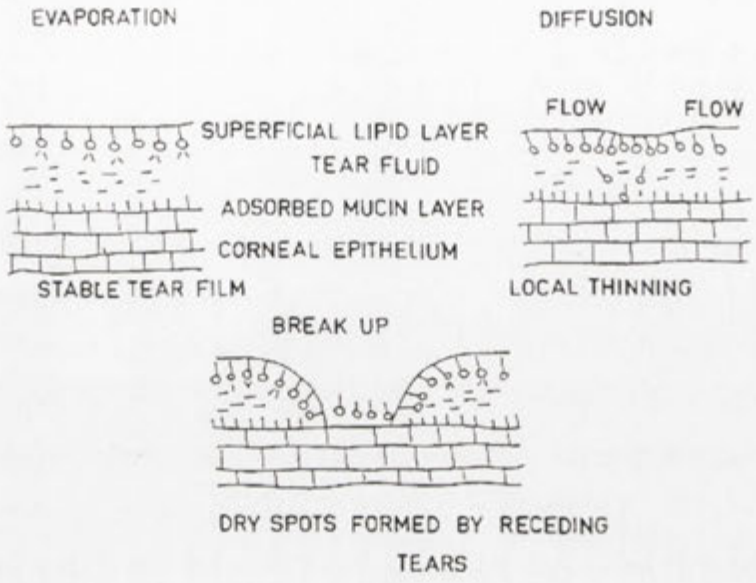


Figure 2.6: *Proposed mechanism of dry spot formation. Courtesy of Holly, 1973 [46]*

2.3.2 Current clinical techniques to study the tear film

A large number of clinical, experimental and research techniques have been developed to study the tear film. In this chapter I review very briefly some of the more prominent techniques and focus on the Fluorescein Break Up Time (FBUT), which is the method we have used to assess the tear film in our research. A full summary of the techniques appears in the DEWS report [18] and in [68].

No "Gold Standard" exists for the diagnosis of dry eye. Therefore, when a test is being evaluated for efficiency, the test population might have been classified using those same tests. Similarly, assessing the performance of new test can be compromised based on the selection criteria of dry eye.

Most of the tests are defined using cut-off points, which differentiates between a dry-eye patient and a non dry-eye patient. For example, a threshold can be the number of seconds passed or the total scoring in a questionnaire. This clearly makes the decision very rough. Correlating it to image processing techniques is equal to the assumption that we have a single threshold (for each test) for an image that separates the object from the background. As it is usually does not perform well in image pro-

cessing applications, it does not work well for dry eyes. Therefore, there is a need for a more sensitive method. For example, a method to distinguish between the different degrees of thinning of the tear film will be useful.

Patient history

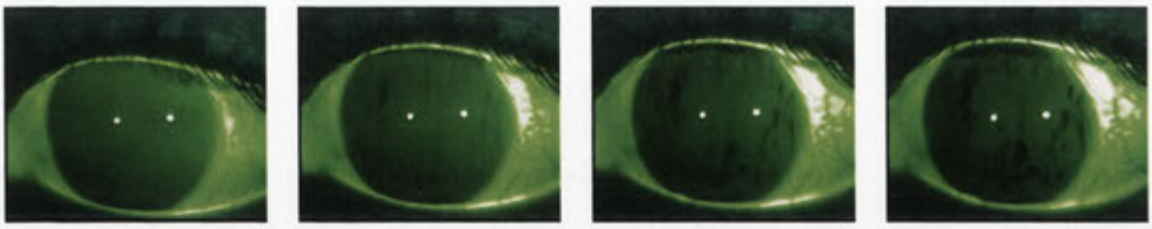
The presence of symptoms needs to be ascertained before any examination. Symptoms might vary in type and severity according to the state of the tear film. The use of a questionnaire is beneficial as it allows the grading of symptoms and is repeatable for comparison purposes. The most common questionnaire is the McMonnies questionnaire. In this questionnaire, a value is assigned to each question and the total score obtained is an indication of the severity of the symptoms. These questionnaires should be used in combination with objective clinical measures of dry eye status.

Tear Flow - the Schirmer test

The tear flow rate was first measured and referred to as tear flow in 1903 by Schirmer. A strip of filter paper is placed in the lower conjunctival sac and is used to collect the tear secretion. The test is performed without anesthesia with the patient's eyes closed. The diagnostic cut-off used was less than $5mm$ of secreted fluid in 5 minutes for dry eyes. However, the test has a few different versions with different cut-off times (for example, [8]). There is a wide intra-subject, day-to-day variation in the test measurement, which does not make it too reliable. Regardless of all of its disadvantages, this test is still widely used by ophthalmologists.

Break Up Time measurement

As explained before, the pre-ocular tear film (POTF) in humans does not remain stable for long periods between blinks. Observation of the POTF before a subsequent blink is the most commonly used test of tear film stability. Reduced tear film break-up time is one of the main signs of an abnormal tear film. The tear film break-up time test was proposed by Norn in 1969 [91]. A moistened fluorescein strip is applied to the conjunctiva, and after a couple of blinks to spread the fluorescein evenly, the tear



(a) (b) (c) (d)

Figure 2.7: *The development of dry spots. (a) Image just after the blink (b) Image 3s after the blink (c) Image 8s after the blink. (d) Image just before the next blink, 11.2s after the blink*

film is viewed with the help of a yellow filter in front of the illumination of the slit-lamp biomicroscope (as a result, the eye images are tinted in a green color, as can be seen in the images throughout the thesis). When a dark area appears in the uniform coloration, it represents the rupture of the tear film and the time elapsed since the last blink is recorded as the **Break-Up Time (BUT)**. This test is known as the **Fluorescein Break-Up Time (FBUT)** .

The reason that these spots are becoming darker is due to the lack of fluid. Just after the blink, the aqueous layer is intact, therefore containing fluid. The mixing between the fluid and the fluorescein and the usage of a yellow filter makes the iris's area look green. As time passes and the eyes stay open, the fluid in the aqueous layer starts to evaporate. In these areas the fluorescein has less fluid to mix with and dry spots form. A break is basically the complete lack of fluorescence at this point. The deeper the break, the greater the chances of ocular surface damage. If the eyes are kept open, the area of the break will increase in size and breaks may appear in new areas over the cornea. If the film ruptures repeatedly in the same spot a superficial epithelial abnormality must be suspected [74]. If the break up occurs over the center of the cornea, a decrease in visual acuity will be induced [111]. We also distinguish between a **full thickness break**, which is called a break-up in the relevant literature, and a **thinning of the tear film**, which is an area in the cornea that starts to rupture, but has not yet fully ruptured, thus it is not a full thickness break.

Fig. 2.7 shows how the dry spots form after the blink. The first image is the image just after the blink, the second image is 3 seconds later and one can see how dry spots

form in the left side and near the upper meniscus. However, these areas are probably showing only thinning of the tear film and not a full break. The third image is 8 seconds after the blink, and the dry spots form all over the cornea. Most of the dry spots are definitely full thickness breaks. The last image is just before the next blink, and the dry spots are becoming darker, a symptom of a full thickness break in these areas.

Many factors affect the measurement of the BUT. Those reducing the BUT include the forced blinking before the measurement and holding the lids forcibly open during measurement. Those increasing the BUT include the decrease in the fluorescein concentration. Various concentrations and volumes of fluorescein has been suggested for the test. In addition, the invasiveness nature of the exam adds to its subjectiveness, creating too many uncontrolled variables. The established cut-off for dry eye has been suggested in 1973 and is 10 seconds [75]. Recently, it has been suggested to use a cut-off of 5 seconds [1] when small amounts of fluorescein are instilled. However, at present, there is a lack of data to support this threshold. The DES subcommittee suggests that refinement of this kind of data would comprise a welcome addition to the literature. We show later in the thesis that by using our new image based definitions for the BUT, we are actually rephrasing the problem and making the measurement much more robust.

Korb in [68] also points towards some of the limitations of the FBUT test. The usual rationale to explain the FBUT test's lack of reproducibility is in its inherent limitation as an invasive technique, thereby altering the tear film stability. The main factor is the volume of fluorescein used. Controlling the amount and concentration of fluorescein instilled is not easy, leading to additional variability and compromising the reproducibility of the results. The volume of the tear film of the eye is only $7\mu l$, while the volume delivered by the strip is over $17\mu l$. Therefore, adding over twice the total tear volume to the eye for the evaluation of FBUT destabilizes the tear film and compromises the measurement. An improved FBUT test would require utilizing the smallest possible controlled volume of fluorescein, minimizing the primary variables: disruption of the tear film and control of volume.

FBUT modification - the DET test

In a report provided by Marquardt et al. [85], it was found that a volume of $1\mu l$ of 2% fluorescein solution instilled into the tear film provides a more accurate and repeatable method of measuring the BUT. Recognizing the need for a new strip that will deliver this exact amount, Korb et al. [69] have developed a new strip called the DET Test. The new strip always delivers the same amount of fluorescein and accomplishes that without sensation and disruption of the tear film. The adequate amount of fluorescein is achieved immediately after instillation and it lasts for $1.5 - 5$ minutes, depending on the patient.

Korb summarizes in [68] that the FBUT, although not ideal, is the best clinical test currently available for identifying tear film instability. Using the DET test produces much more reliable and repeatable results as the amount of fluorescein instilled is as small as possible and fixed. The FBUT test (with or without DET) is the first choice of diagnostic test used by clinicians as almost every clinic has a biomicroscope.

Non-Invasive break-up time (NIBUT)

NIBUT measurement utilizes a grid or other pattern directed onto the pre-corneal tear film for the observation of distortion of the grid in the image. The time since the blink until the change of the grid is called the NIBUT. This method eliminates the need to instill fluorescein and the possibility of reflex tearing. The Tearscope Plus [38] is such an instrument, that is used by inserting a grid into a tube. Therefore, different grids can be used to detect different symptoms. From a medical imaging point of view, such a device is advantageous, as the grids can be altered to best fit the needs of the software engineer. Fig. 2.8 shows the images produced from the Tearscope. In parts (a) and (b), we have used a circular grid with intersecting lines. In part (c), the grid consists of concentric circles, and other grids can be designed easily. The deformation of the grid is related to the appearance of dryness symptoms and the instability of the tear film.

The NIBUT test might seem as the ideal test from a theoretical point of view to measure tear instability. However, it was reported to be in use by only 5% of the

(a) (b) (c)

Figure 2.8: *The Tearscope for NIBUT. (a) Perfect grid shows that there are no abnormalities in the tear film. (b) The deformed areas in the grid are areas of tear film instability. (c) Using a different grid and depicting a major disturbance to the tear film*

clinicians, as the instruments are not always available. The NIBUT values are two to four times higher than FBUT values, and the reason for that has not been completely explained. The NIBUT test has a role in tear film diagnosis, but yet cannot replace the FBUT.

2.3.3 The Eye-Scan device

The Eye-Scan enable users to capture digital images, both still and video, for screening of diseases and disorders on the cornea and the retina. The device is small (see Fig. 2.9(a)) and data can be uploaded via the Internet and transmitted for expert diagnosis. Patients can receive high quality health care regardless of their geographical location. The main part is the camera handle, which contains a digital CCD camera, a camera lens and controls for the illumination. The necessary optical parts as well as the illumination systems are included in the modules, which are clipped on the camera handle (see Fig. 2.9(b)). The removal and clipping of the modules on the handle can be performed in a matter of seconds. The following modules can be clipped on the Eye-Scan:

- Non Mydriatic Fundus Module (NMFM) - It is used for examination of the deeper structures and the fundus of the eye. No medical dilation of the pupil should be necessary.

- The mydriatic fundus module - It is used in the same way as the nonmydriatic

fundus module, but a dilation of the pupil with medicaments is necessary.

- Slit lamp module - It enables stereoscopic magnified views of the frontal structures of the eye, and with a dilated pupil, views of the deeper parts are possible. It is used for the diagnosis of cataracts, retinal detachment or macular degeneration.

- Fluorescein module - It is used to detect corneal abrasion, corneal ulcers and dry eye. Topical fluorescein eye drops are applied in the eye, which is then exposed to blue light. The fluorescein is excited by the light and by using a yellow filter, dark spots appear over the cornea (break of the tear film). This module should be used when performing the FBUT test and produces very similar images to those produced by the equipment the clinician uses in the clinic (slit-lamp with the slit covered).

- Fluorescein module with grid - It is possible to project a grid on the cornea in order to determine the topography of the cornea and the tear film. It can be used with or without the instillation of fluorescein. This device produces similar images to the NIBUT device, the Tearscope.

- Red Reflex module - This module uses infrared light to illuminate the cornea, and can be used for the detection of cataract by looking at the reflected shadows. It is also possible to use clear white light instead.

2.3.4 Our clinical routine

In order to be consistent in our experimental tests, we devised a clinical routine that was followed throughout all our experiments. This routine is based on the improved version of the FBUT test using DET strip and the Eye-Scan device. As mentioned before, this is the most commonly used test to evaluate the tear film. All fluorescein images shown in this thesis and the analysis are based on videos captured using this technique.

1. Instill a small amount of fluorescein in the right eye. The amount should be as little and constant as possible. We have used the DET strips when possible,

(a) (b)

Figure 2.9: *The Eye-Scan device. The device is small and can be used hand-held or mounted in the clinic. By connecting the device to a laptop, the images are saved immediately on the computer and ready to be analyzed by the software. (a) View of the back of the device, where the illumination and the focus can be controlled. (b) Demonstrating the relatively small size of the device. The module used here is the tear film module (the blue piece attached to the front of the device). Using different modules just requires the removal of this piece and the attachment of the new module.*

which have a fixed amount of fluorescein. Otherwise, we used the regular fluorescein strip and cut it so it produces a minimal amount of fluorescein.

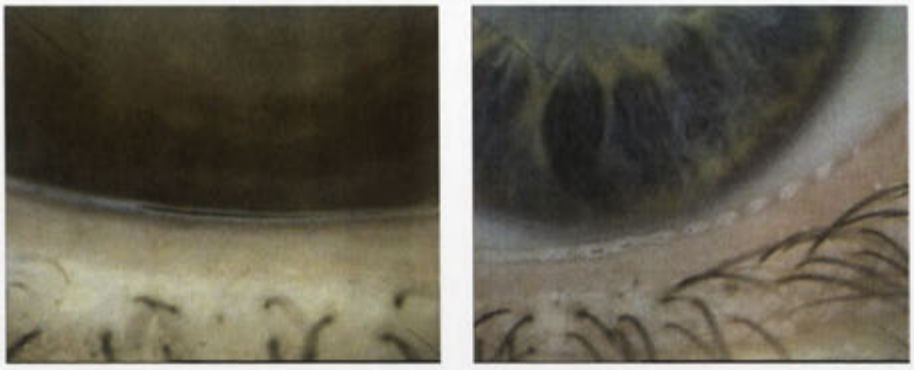
2. Instruct the patient to keep his eyes open as long as possible. However, no longer than 30 seconds is required, as any longer has no clinical significance.

3. Use the Eye-Scan fluorescein tear film module to record the video from the blink until the next blink. By using the Eye-Scan the amount of light emitted and the angle of view can be controlled, making these parameters fixed when used by different clinicians (However, as the camera is hand-held and the specific light conditions cannot be controlled, these parameters still vary).

4. Ask the patient to have a couple of full blinks and repeat the pervious step once more.

5. Repeat steps 1-4 for the left eye.

The whole process takes only 2 to 3 minutes for both eyes and results in a total of 4 sequences. As such, it can be done easily by the clinician for every patient.



(a) (b)

Figure 2.10: *Viewing the lower meniscus using the Tearscope. (a) A thin regular meniscus, as the central black line is continuous. (b) An irregular meniscus. The central black line is deformed and consists of a few parts.*

We call all the frames between blink to blink **A Sequence**. Therefore, our input is 1-2 sequences for each eye that can be used to analyze dry eye. We also use static images for the analysis of the meniscus. The image within two seconds after the blink is taken, as the meniscus shape tends to stabilize at that time, and provides the most information.

2.3.5 Observing the tear meniscus

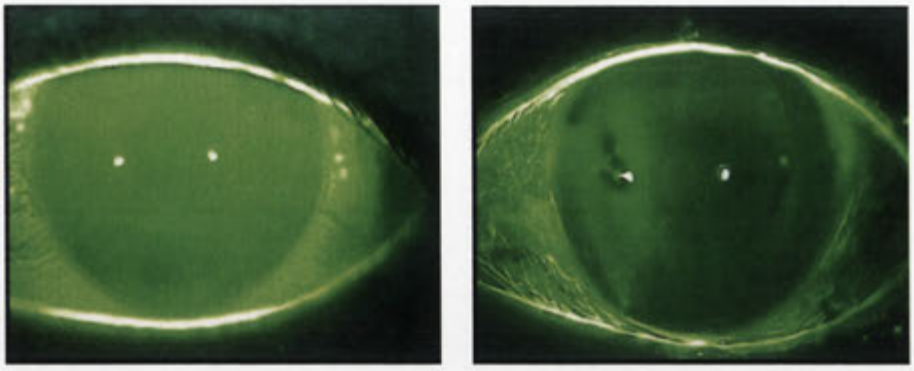
As explained in section 2.1.1, up to 90% of the tear volume is found in the superior and inferior tear menisci. Therefore, the observation of the lid meniscus height and its irregularity has been proposed as a guide to the diagnosis of dry eye. Holly and Lemp [49] suggested that a scanty appearance or an area of discontinuity are signs of an aqueous tear deficiency. Lamberts et al [72] found that the tear meniscus height in normal eyes varied between $0.1 - 0.3mm$ in 92% of their observations and there is no correlation between the height and the Schirmer test. Mainstone et al found [84] that a shallow or irregular meniscus is highly significant because it often results from the many risk factors associated with dry eyes. They concluded that the tear meniscus assessment can be a useful alternative to existing tests for dry eyes.

Analysis of the tear meniscus is an area of recent interest as different imaging modalities are invented and used. The most common methods are probably bio-

microscope based techniques. The Tearscope Plus [38] illuminates the tear meniscus along its whole length and permits the non-invasive observation of morphological changes during the blink sequence. In the central area, the tear meniscus presents a black central band bordered by bright bands on the lid side and tear film side (see Fig. 2.10(a)). Any irregularity of the meniscus shape can be seen by a distortion of the central black band (see Fig. 2.10(b)). Any irregularity in the tear meniscus height can also be observed along its length [113]. With practice, the clinician can grade the tear meniscus. However, it is not an easy task as the height can vary along the meniscus. When the clinician measures the meniscus height, it is usually performed over the corneal area only. A different non-invasive method that measures the meniscus curvature and correlates it with tear volume is meniscometry [124].

The tear meniscus can also be analyzed after the instillation of fluorescein, which will highlight the meniscus. When using the DET test, there should be a minimal increase in the volume of the meniscus. Fig. 2.11(a) shows an image of a regular tear meniscus. Both the lower and the upper menisci have a fairly constant height, which is considered to be regular. Also, there is no irregularity in the shape. In Fig. 2.11(b), there is a major change in height between the left side and the center and the right side. This is a symptom of an irregular meniscus, and can be related to dry eyes. When we evaluate the tear meniscus later in the thesis, we use this kind of fluorescent image and include this measurement in addition to the tear film analysis.

It has also been suggested that the presence of lip like folds of the inferior conjunctiva is a sign of tear film related ocular surface problem called Conjunctivochalasis (CCH) [89]. CCH is usually associated with older people and is sometimes overlooked as it is regarded as a normal aging variation. The folds associated with CCH are called Lid-parallel folds or conjunctival folds [30] and are a sign of loose bulbar conjunctiva. Folds can sometimes be mixed with dry eye and is the predominant diagnosis when dry eye cannot be managed by conventional treatment. The folds are graded for severity with four grades: grade 0 - no permanent folds; grade 1- individual small folds appear in the primary position lower than the tear meniscus; grade 2 - multiple folds appear up to the height of normal tear film meniscus; and grade 3 - multiple folds appear higher than normal tear film meniscus height. It has been



(a) (b)

Figure 2.11: *Viewing the meniscus after the instillation of fluorescein. One can see how both the lower and upper menisci are highlighted due to the fluorescein. (a) Regular meniscus. (b) Irregular lower meniscus with conjunctival folds grade 3. These folds will be hardly visible when using other techniques*

demonstrated that patients with folds have a higher risk for dry eyes, sometimes up to 190 times if grade 3 folds exist. Fig. 2.11(b) shows a lower meniscus with grade 3 folds in the central area. These folds cannot be detected when using a bio-microscope and usually require the instillation of fluorescein in order to be evaluated correctly.

It seems that the folds disturb the morphology of the meniscus and its relationship with the lid edge. In presence of folds, the reservoir morphology is abnormal, thus limiting the uptake of tears during the blink and reducing the resurfacing of the POTF over the conjunctival surface. Such alterations increase the signs and symptoms of dry eye. Moderate and severe folds can cause pain and ulcer subconjunctival hemorrhage.

Another emerging technique for viewing the tear meniscus is using Optical Coherence Tomography (OCT). OCT, in contrast to fluorescein imaging, is a non-invasive and non-contact technique which provides a cross-sectional image of the retina at a specific location. Such a cross-section is called an A-Scan. A single A-Scan shows the different layers of the retina at that location. A set of A-Scans is called a B-Scan. For example, a circular scan around the optic disc produces a set of A-Scans showing the different layers of the retina in a region around the optic disc. Scans around the optic disc can aid towards the detection of glaucoma by examining the thickness of the retinal layers at different spatial locations. Scans of the macular region can aid towards the detection of AMD (Macular Degeneration).

Obviously OCT images are very different from those produced using a bio-microscope. However, there are attempts to use it to view the tear meniscus [9, 92, 98]. To that end, a vertical scan is taken over the lower or upper eyelids region (or both) and the cornea. Such a scan shows the tear film, cornea and eyelids. When using high magnification, the tear meniscus is also visible, depending on the specific location of the scan. Therefore, a collection of B-Scans in the tear meniscus region can be used to compute the tear meniscus height. Another interesting observation that can be estimated from B-Scans is the depth of tear meniscus. The depth varies at each B-Scan and can be used to estimate the curvature of the meniscus. In addition, the area of the tear meniscus can be computed. However, this information is computed for that specific location only, since the images are sagittal cross-sections through the eye – quite different from our frontal images. In order to estimate the full shape of the meniscus, many B-Scans have to be taken at different locations. Then using an interpolation technique, the full shape can be recovered.

Currently, OCT imagery is mainly used for viewing the fundus as the device is not available to most clinicians. In addition, it is much more expensive than a slit lamp and conjunctival folds cannot be detected. The above reasons make the routine use of OCT by clinicians for detecting DES uncommon, favoring a slit-lamp based solution. Nevertheless, the OCT device has the advantage of being a non-invasive technique and that the volume of tear meniscus is not altered due to the fluorescein.

Chapter 3

Previous Work

Existing research is mainly oriented towards medical ways of improving the reliability of the tests used for detecting the tear film quality [31, 45]. It includes non-invasive tests using devices such as Optical Coherence Tomography (OCT) and other dedicated devices [35, 38, 54, 116] and also proving the relationship between the area of break and dry eye [7, 10]. Some of these devices and methods have not been proved to be clinically useful by testing on large number of patients and others are not available to clinicians. For example, the OCT device is far less prevalent than a slit-lamp and also more expensive. Furthermore, most of these methods do not directly relate to our research as the imaging modality is very different and they have no automatic components. On the other hand, the FBUT test has been used clinically for a long period now and can be carried out in almost every clinic.

There are some existing approaches to locate the iris as part of other applications, mainly for iris recognition. Usually, assuming its shape is a perfect circle, the methods mostly use circle fitting algorithms to first locate the pupil and then the iris. Recent methods drop the assumption of circularity of the iris and also handle eyelashes and shadows.

The chapter is organized as follows: first, iris detection methods are discussed and I will demonstrate the difficulties of applying some of the traditional methods on our eye images. Then alignment algorithms are discussed, as the alignment is a crucial step in our dryness algorithm. Finally, I will briefly discuss non-automatic methods for the detection of dryness. These methods are based on dedicated hardware or on the clinician manually segmenting areas of interest. As non-automatic methods and clinical hardware are not the focus of the thesis, only a short discussion is provided.

3.1 Iris Detection

Quite a few techniques appear in recent literature for iris and pupil detection. Most of these techniques are for iris recognition, which is used for personal identification. Probably the most known method has been published by Daugman [29] in 1992. Since then, he has revised the algorithm in further publications. Daugman initially employed a circle fitting method for both the iris and the pupil, which was based on computing the first derivative along a circular arc. His method is discussed in more detail in the next section. Another early iris segmentation algorithm was suggested by Wildes [120] in 1997. His algorithm also uses edge detection methods and fits a circle to the iris and the pupil. The fitting of the circles is performed by using the Hough transform and in addition the upper and lower eyelids are detected. Circle fitting approaches and the Hough transform became a popular trend in further publications such as [32, 78, 107].

Other methods also detect the eyelids and eyelashes to provide a better boundary estimation for the iris and to remove reflection noise [43, 51, 67]. The removal of reflection is important for the recognition step. The eyelids are usually modelled by a parabola and reflections are detected using thresholds and histogram information.

Trucco and Razeto [108] detect the iris to perform automated iris tracking within medical applications. They model the iris as an active ellipse which is minimized by using simulated annealing. However, looking at their examples, the images include large portions of the face, where the iris area occupies less than half of the image's size.

3.1.1 Daugman's method

Daugman focused on iris recognition and finds the pupil-iris and iris-sclera borders as a first step [29]. Given an input image $I(x, y)$, the iris is first detected by using similar ideas to a circular edge detector. The iris is defined as a circle and the aim is to find the best parameters (x, y, r) which maximize the difference in intensities between pixels on the contour of the iris. This process is done at different scales by convolving the image with a smoothing Gaussian G_σ starting from a coarse level to a fine level. Given

the intensity image $I(x, y)$ and a circular arc ds of radius r, the continuous problem is defined as finding the maximum energy of the following cost function:

$$max_{(x,y,r)}|G_\sigma(r) * \frac{\partial}{\partial r}\oint_{(x,y,r)} \frac{I(x,y)}{2\pi r}ds| \tag{3.1}$$

The partial derivative $\frac{\partial}{\partial r}$ computes differences in intensities between pixels and is similar to computing gradient information along the arc ds. The function $G_\sigma(r)$ is a smoothing Gaussian which is convolved with the image. Therefore, for a given center coordinates (x, y) and a radius r, the sum of differences in intensities is computed along the contour. The aim to find the center coordinates and radius that maximize this sum. The search for a given center coordinates starts with a radius r which is gradually increased. The whole process is performed at successively finer scales of analysis starting from the coarsest level - high smoothing to the finest level - least smoothing.

The pupil is then found in a similar way, using only a confined search area. In the original paper [29], the circular arc ds do not span a whole circle, but is limited to two opposing 90 degree arcs centered on the horizontal meridian (so only part of the circle is taken). The reason is that the eyelids usually cover parts of the iris both in the upper and lower parts, thus a full circle is not actually visible. These arcs are depicted in Fig. 3.2). However, this limitation is not mentioned in some newer publications, such as [27].

In order to speed up the computation, the discrete implementation of (3.1) is used. In addition, the order of the smoothing operator (the Gaussian $G_\sigma(r)$) and the differentiation (the $\frac{\partial}{\partial r}$ operator) is changed and they are concatenated to create a new operator. The discrete convolution is computed last between this resulting operator and the sums of pixels along the contours of the arcs. Thus, the cost function that is actually being maximized is the discrete form of:

$$max_{(x,y,r)}|\frac{\partial}{\partial r}G_\sigma(r) * \oint_{(x,y,r)} \frac{I(x,y)}{2\pi r}ds|. \tag{3.2}$$

The discrete forms of the image integral and convolution are used for the computa-

tion. The partial derivative is implemented as a discrete form of a first derivative. It uses Δr, which is defined as a small increment in the radius r. Therefore it is just a difference between two elements defined by the interval Δr:

$$\frac{\partial G_\sigma(r)}{\partial r} = G_\sigma^{(1)}(n) = \frac{1}{\Delta r}[G_\sigma(n\Delta r) - G_\sigma((n-1)\Delta r)]. \tag{3.3}$$

Finally, the maximization process is further sped up by using a multi-grid approach (solving a coarse problem and then refining by using finer grids). This approach uses large increments of Δr in the coarse scale and the increments are decreased in successively finer scales. A similar idea is also used for refining the coordinates of the pixels that are used for the computation of (3.2). As a result, in finer scales, the search is performed in increasingly smaller areas. This search is performed by using gradient ascent (as we are maximizing a function).

Daugman's method has been reliably used on eye images for a long time. It also produces good results on our images for the detection of the iris. However, it is not fully usable to detect the iris in our fluorescein images due to three main reasons: (1) The pupil is not visible in our eye images (2) fluorescein images tend to have more noise than Daugman's eye images and the borders between the iris and the conjunctiva are fuzzy (3) the method has been patented. I will now demonstrate the second issue. In the eye images we use, quite often the eyelids have the strongest edges in the image because of the strong response to the bright fluid found in them. Moreover, the fluorescein spreading makes the borders of the iris fuzzy and noise on the conjunctiva is more evident because of the high magnification. These ideas are illustrated in Fig. 3.1.

The first row in Fig. 3.1 shows that even in the most coarse level (or high smoothing), the eyelids are those with the strongest gradients and the biggest differences in intensities occur between the eyelids and the conjunctiva or between the eyelids and the iris. Therefore, Daugman's method might detect those circles that do not define the borders of the iris. The second row shows only the higher magnitude pixels of the image (using a reasonable threshold). These pixels are of interest as they produce the highest gradient difference over the iris contour. Here, the whole 360 degrees of the

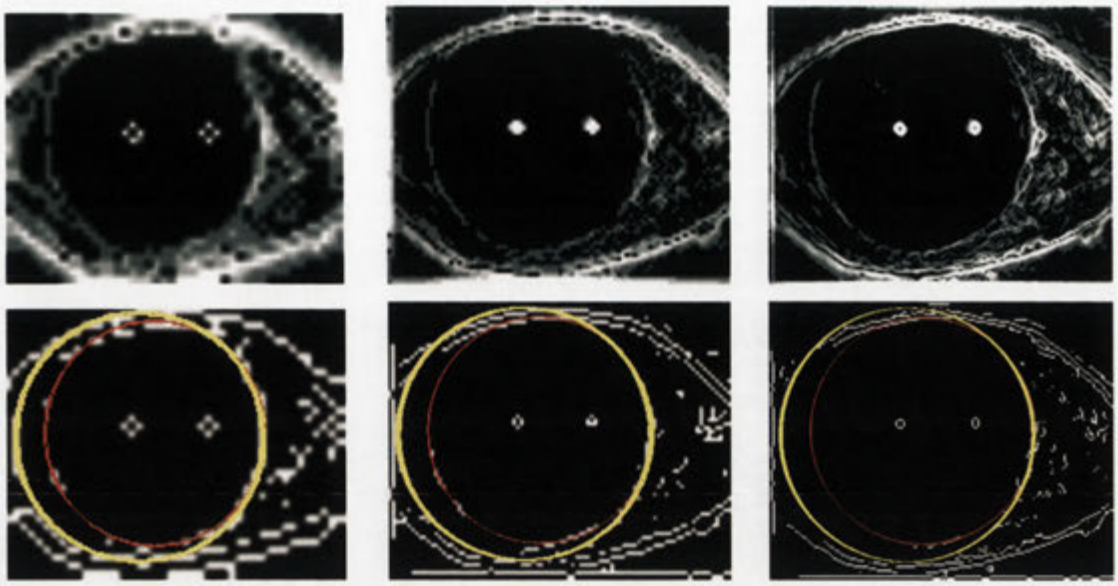


Figure 3.1: *Edge information of an eye image in different scales. Images at column (a) are scaled down 8 times, at column (b) 4 times and column (c) 2 times.* ***First row****: Magnitude image using 4 different grey level intensities. The brightest pixels correspond to the highest magnitude. Pixels with a low magnitude appear as black. The brightest pixels are mainly parts of the eyelids even at the smallest scale.* ***Second row****: Edge image after a threshold. The red circle corresponds to a good fitting of the iris and the yellow circle shows a solution that has a higher cost (using Daugman's method) than the optimal fitting for the iris. (Note that in the right hand side the two circles merge). At the largest scale the circle does not fit to any pixels on the left hand side and since only small changes in Δr are allowed at finer scales, the correct fitting cannot be found.*

iris circle are used for the computation of (3.1). A possible correct fitting of the iris with a high cost when using (3.1) is shown in red. In yellow, a fitting with a higher cost than the fitting in red is shown. The chosen threshold depicts why such fitting has a higher cost. Almost all the pixels of the iris in the left hand side have weaker edge information than the eyelids or the noise. Therefore, the eyelids make the strongest contribution to (3.1). We note that using only a 90 degree opposing arcs would provide an accurate fitting for this example.

In Fig. 3.2, we have used only 90 degree opposing arcs for the computation of (3.1). The upper eyelids are not fully open and therefore even the partial arcs contain portions of the high magnitude eyelids. In the first row of the figure the eye image is depicted in three different scales and the correct fitting is plotted on top. The 'x'

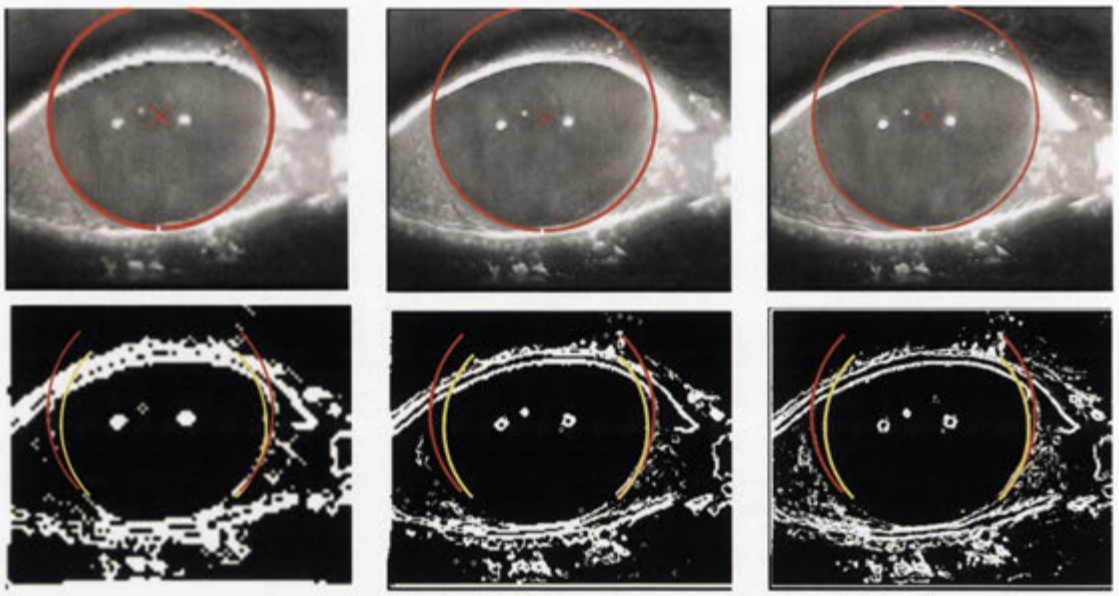


Figure 3.2: First row: *An image of the iris taken from a video shown at three different scales: (a) image scaled to 1/4 of the size (b) image scaled to half the size (c) the original image. The 'x' corresponds to the center of a correct fitting to the iris.* **Second row:** *A threshold image of the corresponding image from the first row showing only higher magnitude pixels. The red circles are the correct fitting to the iris and the yellow circles show a solution that has a higher cost than the fitting in red (using (3.1)).*

is the center coordinates of the iris and it can be observed that it is fairly close to the upper eyelids. In the second row of the figure, a threshold image is depicted (in a similar way to Fig. 3.1, only the higher magnitude pixels are shown). The 90 degree arcs corresponding to the circles from the first row are plotted in red (the location of the arcs is computed using the center coordinates). The yellow arcs correspond to a fitting with a higher cost when using (3.1) than the optimal fitting in red. The difference in cost of (3.1) between fittings is less noticeable in the highly smoothed image, but becomes significant in favor of the smaller (yellow) circle in the large scale. As there are only a few high magnitude pixels on the left side of the iris, the cost function will try to use as many as possible pixels from the eyelids to increase the sum of differences.

Daugman's method does not face these problems in his images, since a good estimate of the pupil can be found. However, in our images, the pupil is not visible due to the instillation of fluorescein. Assuming the pupil and the iris are roughly concen-

tric, it is possible to limit the search area for the center of the iris. Therefore, it can immediately eliminate most of the erroneous estimations.

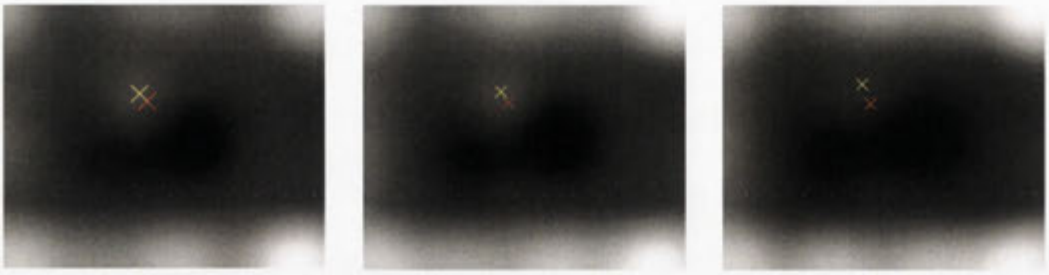
3.1.2 Additional approaches to circle detection

Ma et al. [83] also focus on iris recognition and they detect the iris in the first step. Their implementation starts by finding the center of the darker pupil simply by looking for the darkest row and column in the image. Then the estimation is improved by searching a localized region around the estimated location of the pupil. An adaptive threshold based on the gray level histogram of this region is selected. Then the center of the binary region is obtained by using the center of all pixels below the threshold. The pupil's exact coordinates and radius are found by using the Canny edge detector and the Hough transform. This process is repeated for the iris detection as well. The center coordinates of the iris can be different from those of the pupil (not concentric), but they are estimated using the pupil's center coordinates.

This application is based on detecting the pupil, which provides a good estimation for the center coordinates of the iris. The Hough transform is performed only over a small search region by approximating the iris region and is fairly fast. The Hough transform offers a potentially good solution, but we struggled to get it working on our images due to its difficulty in detecting the iris when only partial edges are visible and its tendency to be overly affected by noise. Moreover, as the search is performed with varying radii, it is not easy to know which fitting is the best.

Fast Radial Symmetry [81] is a technique to detect circular objects in an image. The method finds strong gradients in opposite directions and picks the gradients according to a strictness parameter. The range of radii to be searched has to be supplied to the method. According to the examples in the paper, the method seems to produce the best results when detecting circles of a small radius, such as eyes in a full facial image.

When applied to our eye images, the method worked quite well in some cases, especially when the eyelids cover a small portion of the iris or when it is fully visible. The method's output provides the center of the circle, but not the radius. Since a



(a) (b) (c)

Figure 3.3: *Images produced by applying the Fast Radial Symmetry to the images from Fig. 3.2 using three different scales: (a) 1/4 of the original image (b) half of the original image (c) original image. The red 'x' corresponds to the correct location of the center of the iris and the yellow 'x' corresponds to the best estimation found by the Fast Radial algorithm.*

large range of radii is given to the algorithm as an input, the correct radius can be anywhere within this range. As in our eye images (in the finest scale), the radius ranges approximately from 96 to 144 pixels, an additional search has to be applied to actually find the correct radius.

However, when we applied the fast radial symmetry algorithm to the image from Fig. 3.2, the result was less good, as depicted in Fig. 3.3. In the figure, we show the inverse image provided by the algorithm at three different scales. The darkest regions should correspond to maximum symmetry as the gradient is calculated from dark to light. The red 'x' marks the correct center of the iris and the yellow 'x' corresponds to the minimum value found by the algorithm (when limiting the search to only the center of the image). The algorithm performed the best at the coarsest scale, but the error in estimating the center is quite evident at the finest scale (see Fig. 3.3(c)), being shifted by 20 pixels in the vertical axis from the real center. This is probably due to similar difficulties Daugman's method faces. In addition, in the finest scale, the algorithm is not so fast any longer unless we limit the number of different radii given to the algorithm.

To sum up, we have found that it is better to first detect the upper and lower eyelids, which have a strong gradient information and remove them. Then the chances of a correct detection of the iris are higher. Additionally, due to the possibly weak gradient information of the iris, its detection should not be based solely on magnitude of

the pixels.

3.1.3 Active Contours

Other approaches to detect the iris use active contours and usually do not assume a circular shape for the iris and pupil [2, 4, 28]. Daugman in a recent work [28] extends his previous work by removing the assumption of circularity. He notes that that the iris and the pupil are not concentric and the pupil boundary is non-circular and the iris boundary is usually non-circular as well. In addition, the iris is often occluded by eyelids and sometimes by reflections. Therefore, it is necessary to fit a flexible contour that can tolerate interruptions.

Daugman presents curvature maps for the inner (pupil) and outer (iris) iris boundaries. The curvature map should be flat and straight if the boundaries were circles, as the curvature of a circle equals to the inverse of its radius everywhere. However, in his examples, the inner and outer boundaries are definitely not flat. The curvature map of the iris tends to be flatter than the curvature map of the pupil. In other words, the iris is closer in shape to a perfect circle than the pupil.

His solution is based on active contours and using Fourier series expansions. A discrete Fourier series approximation is fitted to the data that is based on samples of the radial gradient data (he assumes that an initial detection of the boundaries is provided). A set of N regularly spaced angular samples is used to compute each coefficient. A set of M discrete Fourier coefficients are taken to provide an approximation to the inner and outer boundaries. This creates a new sequence of N pixel's locations, but this time they do not necessarily represent a circle. The value of M defines the tradeoff between how precise the fitting is versus keeping the model simple. He used $M = 17$ and $M = 5$ to estimate the pupil and the iris boundary respectively, since the gradient information of the iris is much weaker.

He et al. [42] present a fast algorithm for iris detection that, according to their findings, is faster and superior to Daugman's [29] and Wildes' [120] algorithms. They

demonstrate that the latter algorithm can be stuck in local maximum when locating the iris.

The iris is first detected using a learning process where the best features are chosen by using the AdaBoost algorithm. The learnt iris detector searches over windows with different scale and position and validates whether the computed features match those defined by the AdaBoost as representing an iris. This provides a rough position for the center of the iris and also rejects non-iris images (however, in our application all images should contain the iris).

The second step of the algorithm is the interesting one. Given an approximation of the center of the iris and its radius, edge points are detected. The detection is performed in polar coordinates, as a circle in polar coordinates is a line. The image is transformed to polar coordinates centered by the center of the iris and a vertical edge detector is used. One edge point is reserved along each column and each point is associated with its distance from the approximated center of the iris. It is not mentioned why specific edge points are chosen while others are rejected.

The authors developed a method of pulling and pushing based on Hooke's law. They define an iterative process by using springs: given a set of edge points, each of them is connected to the center of the iris using springs. At the ideal situation (equilibrium) there are no forces applied on the springs. Therefore, the sum of forces on springs that are located at opposite radial locations is 0. Then a force is exerted on the center of the iris to move it to a new location and as a result it deforms the springs. This force is applied on each spring according to its distance from the center of the iris (which has been estimated before). At this stage, the sum of forces on springs on opposite radial locations of the circle will not necessarily equal to 0. The total sum of these forces (vectors) defines the displacement towards the new center. The algorithm then iterates with a new set of edge points until convergence. Although this process is defined using springs, it is closely related to active contours and also resembles ideas from the fast radial algorithm (however, the gradient's magnitude is not used here).

The authors extended their algorithm in a very recent publication [44]. This time they remove the requirement that the pupil and the iris are circles. The pushing forces can result in an ellipse shaped iris or pupil, therefore increasing the tolerance to pupil

distortion.

The paper also provides a method to detect the eyelids. The authors define this problem as "An even harder problem" than the iris fitting, even though they only segment the eyelids in the iris region. The flowchart of their algorithm consists of the following steps: 1) Detecting and removing eyelashes by using a horizontal edge detector; 2) Using a vertical edge detector to detect the eyelids, as the eyelids usually have the strongest gradients in that direction. The authors mention that only one point is reserved in each column for the eyelids, but do not describe how they choose which point. This step creates a raw eyelid image; 3) The raw eyelid image is fitted to the best predefined eyelid model; 4) The detected eyelid points are interpolated according to the model to create a smooth curve by fitting a parabola.

The interesting part of the algorithm is in step 3, where the eyelids are fitted to a predefined model. The authors assume three possible models that describe the shape of the eyelids. Each model depicts the shape of the eyelids according to their height at each column. For example, flat eyelids will have small height changes throughout the columns, while curved eyelids have large changes in height when moving from the leftmost to the rightmost columns. The similarity is measured by subtracting the raw eyelid image from each model. The subtraction result that is closest to a flat line defines the most suitable model. This model is then used to define the parabolic curve fitting. An assumption the authors make in the model creation is that the pupil is centered within the eyelids' region. Hence, if the person's gaze is not straight, no model will necessarily match the raw eyelid image.

Finally, the authors present a method for eyelash and shadow detection. The shadow detection is somewhat similar to our black-line detection problem (described in chapter 5.5.4). The authors divide the iris to two regions: upper region (near eyelids) and the rest. Then they build intensity histograms for the two regions, and check if one is darker than the other. If so, an adaptive threshold is estimated from these histograms to detect the darker eyelids. When examining the images we have in hand, we have found out that this method does not work well for us. First, the region near the upper eyelids can be darker than the other region even if there are no eyelashes due to uneven fluorescein spreading and the angle of the camera. Second, our images

are not necessarily illuminated evenly, so one side of the iris can be brighter than the other. The histograms of the two regions are expected to be similar in this case. However, a shadow 2-3 pixels in height, that spans the length of the eyelid might have similar intensity to pixels in the darker side of the iris and hence will not be removed. We note that such a shadow has similar characteristics to the black line.

Ritter developed in her thesis a method for registering corneal images that are taken using a slit-lamp. These images are captured in a similar way to ours, but the difference is that we instill fluorescein before taking the images. Since fluorescein in not instilled in Ritter's images, the pupil is visible.

The first step of the registration is to find the inner and outer iris borders, or in other words, the pupil and the iris respectively. Once the center of the iris has been found, it is used as a reference point for translating the images to match a reference image. Pixels on the borders of the iris and inside the iris are used as seeds for the final registration of two corneal images using a 7 parameter transformation. In the first step, pixels on the outer border of the iris in one image are aligned to the other by using 4 parameters: translation, rotation and scaling. In the second step, pixels inside the iris are matched by using a 7 parameter transformation: the center coordinates and the radius of the inner iris border and the 4 parameters from the first step of the alignment.

The pupil is modelled as a circle, and a rough estimation can be easily found by looking at the largest dark mass in the image, while limiting the area of search to the center of the image. Then both the pupil and the iris are detected using a discrete circular active contour (DCAC). The idea behind the contour model is to compose a closed connected polygon of n vertices. Then internal and external forces are applied to the polygon and push it towards a perfect circular polygon. At each iteration, internal forces increase the polygon's average radius by δ. This radius is computed as the average distance of each vertex to the center of mass of the polygon. External forces (or image forces) push the polygon towards the center by a force proportional to the difference in intensities between neighboring pixels near the polygon's vertex.

Assuming v_i is a vertex of the current polygon, β is a weighting factor, $F_{int,i}$ and $F_{im,i}$ are the internal and external forces respectively, the contour movement of a vertex i at time $t+1$ is given by the following formula:

$$v_i(t+1) = v_i(t) + \beta F_{int,i} + (1-\beta)F_{im,i}. \tag{3.4}$$

Estimating the next polygon coordinates is an iterative process computed as a function of the iteration number t. The process stops once equilibrium has been reached, defined as having only a very small change of radius over a few iterations.

While the pupil is modelled as a perfect circle, the iris is modelled by using two concentric arcs on either side of the circle. This represents the portion of the border of the iris which is not covered by eyelids. Therefore, the contours used for the detection of the iris are two 60 degree opposing arcs (one sixth of a circle) centered on the horizontal meridian. This is a similar idea to the one Daugman used with 90 degree opposing arcs. Two discrete circular active contours are then used to represent this model and are moved alternately until either stability has been attained or an error condition has occurred.

The main difficulties of this model are to find the correct increase δ and to locate the iris when its borders are fuzzy. As fluorescein is not instilled in these image, the pupil can be easily detected and helps in finding the center of the concentric circles. In addition, the edges of the iris tend to be strong enough to compute a solution based on active contours.

3.2 Image alignment methods

Alignment of the iris over a series of images in a video plays an important role in our dryness detection method. As explained in section 2.3.4, our clinical routine records a video using a hand-held camera. In addition, it is not always possible for the patient to keep his gaze steady, resulting in the movement of the iris. As we describe and demonstrate in chapter 5, the alignment is crucial in order to detect the progress of the dryness over time and to find the BUT (Break Up Time).

Our alignment method is closely related to the method proposed by Lucas-Kanade for optical flow [82, 5]. This method performs a straightforward (brute-force) alignment and thus do not use image features. In the next sections, I will describe this method and provide a brief description of other alignment techniques.

3.2.1 Lucas-Kanade alignment

The Lucas-Kanade (LK) alignment algorithm is a Gauss-Newton minimization method. This algorithm was later called the Additive LK algorithm. The most common use of the LK algorithm is for the computation of optical-flow. Given two images, the LK algorithm minimizes the following sum of squares:

$$\sum_x [I(W(x, y; \mathbf{p})) - T(x, y)]^2. \tag{3.5}$$

In this equation $T(x, y)$ is the original image and $I(x, y)$ is the image we would like to align. $W(x, y; \mathbf{p})$ is defined as the homography (or warp) with parameters $\mathbf{p}$. The warp assigns a new pixel location to each pixel (x, y) in I. The minimization is performed with respect to $\mathbf{p}$ and the sum is computed over all image pixels. For example, if we only consider translation and scaling, W can be defined as:

$$W(x, y; \mathbf{p}) = [(x + p_1)p_3, (y + p_2)p_4]. \tag{3.6}$$

In this example $\mathbf{p} = [p_1, p_2, p_3, p_4]^T$, where $[p_1, p_2]$ is the translation vector and $[p_3, p_4]$ is the scaling vector. Therefore, there are 4 parameters that have to be minimized.

Minimizing the expression in (3.5) requires a non-linear minimizer, as the pixel values $I(x, y)$ are not linearly related to their spatial location. The LK algorithm assumes that we a have an initial estimate for $\mathbf{p}$ and then iteratively minimizes for increments of $\Delta\mathbf{p}$. Therefore, at each iteration the following expression is minimized with respect to $\Delta\mathbf{p}$:

$$\sum_x [I(W(x, y; \mathbf{p} + \Delta\mathbf{p})) - T(x, y)]^2. \tag{3.7}$$

At the end of each iteration the parameters of the warp are updated:

$$\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}. \tag{3.8}$$

We notice that I is a function of two parameters: $I = f(x, y)$. The warp W maps the pixels from coordinates (x, y) to new coordinates (x', y') according to the parameters $\mathbf{p}$. Therefore W is a function of $W(x', y') = g(x, y; \mathbf{p})$. In order to solve (3.7), the term $I(W(x, y; \mathbf{p} + \Delta\mathbf{p}))$ is expanded using the Taylor expansion with respect to $\Delta\mathbf{p}$ about the pixel location (x, y). To compute the first order Taylor expansion, the first derivative of $I(W(x, y; \mathbf{p} + \Delta\mathbf{p}))$ has to be computed with respect to $\mathbf{p}$. Denoting $\mathbf{z} = (x, y)$, the coordinates of the warp as $W(\mathbf{z}; \mathbf{p}) = (W_x(\mathbf{z}; \mathbf{p}), W_y(\mathbf{z}; \mathbf{p}))^T$ and using the chain rule we receive:

$$\frac{\partial I}{\partial \mathbf{p}} = \frac{\partial f}{\partial x}\frac{\partial W_x(\mathbf{z}; \mathbf{p})}{\partial \mathbf{p}} + \frac{\partial f}{\partial y}\frac{\partial W_y(\mathbf{z}; \mathbf{p})}{\partial \mathbf{p}} = \nabla I \frac{\partial W}{\partial \mathbf{p}}. \tag{3.9}$$

As $\mathbf{p}$ is a vector, the terms $W_x(\mathbf{z}; \mathbf{p})$ and $W_y(\mathbf{z}; \mathbf{p})$ can be further expanded by the chain rule. The first-order Taylor expansion of (3.7) with respect to $\Delta\mathbf{p}$ about $\mathbf{z}$ is given by:

$$\sum_x \left[I(W(\mathbf{z}; \mathbf{p})) + \nabla I \frac{\partial W}{\partial \mathbf{p}} \Delta\mathbf{p} - T(\mathbf{z})\right]^2. \tag{3.10}$$

The term $\frac{\partial W}{\partial \mathbf{p}}$ is the Jacobian of the warp, which is computed at each iteration at $(\mathbf{z}; \mathbf{p})$ and therefore is dependent on the current estimate of the parameters. Minimizing (3.10) by taking the partial derivative with respect to $\Delta\mathbf{p}$ gives:

$$\Delta\mathbf{p} = H^{-1} \sum_x \left[\nabla I \frac{\partial W}{\partial \mathbf{p}}\right]^T [T(\mathbf{z}) - I(W(\mathbf{z}; \mathbf{p}))], \tag{3.11}$$

where the matrix H is defined as:

$$H = \sum_x \left[\nabla I \frac{\partial W}{\partial \mathbf{p}}\right]^T \left[\nabla I \frac{\partial W}{\partial \mathbf{p}}\right]. \tag{3.12}$$

The matrix H can be seen as an approximation to the Hessian of the warp. It is dependent on $\mathbf{p}$ in both the Jacobian and the gradient of I which is evaluated at $W(\mathbf{z}; \mathbf{p})$.

The main difficulty in this solution is that at each iteration of the LK algorithm, the inverse of the matrix H has to be computed until convergence. As a result, the LK algorithm can be slow for big images.

The resulting approach is called a direct approach, as it just computes the difference in intensities between the images in the region of interest. Such a method is mostly suitable for videos where the change between images is small or when it is hard to find distinctive features in the image.

3.2.2 Extension of the Lucas-Kanade algorithm

As mentioned in Section 3.2.1, the LK algorithm computes the Hessian matrix H at each iteration. This is the weakest link of the algorithm which consumes most of the computation time. When aligning a set of 100 images in a video, the process can become relatively slow. Faster methods to compute the alignment have been proposed in [5].

These methods offer a different way to minimize the sum of squares of the difference in intensities of (3.5). In this section I will describe briefly the Inverse Compositional Algorithm for image alignment. The main idea in this algorithm is that the Hessian matrix is not computed at each iteration but is constant through the iterative process.

The Compositional algorithm is very similar to the one described in section 3.2.1. The difference is that the algorithm minimizes:

$$\sum_x [I(W(W(x, y; \Delta\mathbf{p}); \mathbf{p}) - T(x, y)]^2, \tag{3.13}$$

with respect to $\Delta\mathbf{p}$ and at each iteration updates the estimate of the warp as:

$$W(x, y; \mathbf{p}) \leftarrow W(x, y; \mathbf{p}) \circ W(x, y; \Delta\mathbf{p}). \tag{3.14}$$

The difference from the additive approach is that the new parameters of the warp are computed as a composition instead of adding $\Delta\mathbf{p}$ to $\mathbf{p}$. The computational cost of the compositional algorithm is similar to the additive algorithm. Even though the

Jacobian is only computed once at $(x, y; 0)$, the Hessian has to be computed at each iteration. This is because it is still dependent on the parameters $\mathbf{p}$, as the gradient of the image I is evaluated at $W(x, y; \mathbf{p})$. We will now show how the Inverse Compositional algorithm overcomes that.

The Inverse Compositional algorithm minimizes the following function:

$$\sum_x [T(W(x, y; \Delta\mathbf{p})) - I(W(x, y; \mathbf{p}))]^2, \tag{3.15}$$

with respect to $\Delta\mathbf{p}$ and at each iteration the warp parameters are updated as:

$$W(x, y; \mathbf{p}) \leftarrow W(x, y; \mathbf{p}) \circ W(x, y; \Delta\mathbf{p})^{-1}. \tag{3.16}$$

The idea behind (3.15) and (3.16) is that the transformation $\mathbf{p}$ is applied to I as before, but the increment is applied to T. For example, let us assume that the warp is just translation: $\mathbf{p} = [x_1, y_1]^T$, and the increment is $\Delta\mathbf{p} = [\Delta x, \Delta y]^T$. Therefore, in the additive case we would transform the image I by: $[x_1 + \Delta x, y_1 + \Delta y]^T$ and these will be the new warp parameters $\mathbf{p}$ in the next iteration. Here the image I is translated by $\mathbf{p}$ while $\Delta\mathbf{p}$ needs to actually translate T towards I by: $\Delta\mathbf{p} = [-\Delta x, -\Delta y]^T$. Therefore, the update of the warp in the compositional approach is done using the inverse of the parameters increment. It can be thought as reversing the roles of the images I and T. If we represent the warp parameters in homogenous coordinates we receive:

$$W(x, y; \Delta\mathbf{p}) = \begin{pmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{pmatrix}, \quad W(x, y; \Delta\mathbf{p})^{-1} = \begin{pmatrix} 1 & 0 & -\Delta x \\ 0 & 1 & -\Delta y \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.17}$$

So we see that the inverse compositional update role of (3.16) works the same as in the additive case.

Similar to the additive approach, we perform a first order Taylor expansion on (3.15) with respect to $\Delta\mathbf{p}$ about (x, y):

$$\sum_x \left[T(W(x, y; \mathbf{0})) + \nabla T \frac{\partial W}{\partial \mathbf{p}} \Delta\mathbf{p} - I(W(x, y; \mathbf{p}))\right]^2, \tag{3.18}$$

and assume that $W(x, y; \mathbf{0})$ is the identity warp. The important observation here is that the Jacobian $\frac{\partial W}{\partial \mathbf{p}}$ is evaluated at $(x, y; \mathbf{0})$ as this is where the Taylor expansion was performed. Therefore, the Jacobian is a constant that can be computed once. This is similar to the Compositional algorithm. The solution to the least squares problem is:

$$\Delta \mathbf{p} = H^{-1} \sum_x [\nabla T \frac{\partial W}{\partial \mathbf{p}}]^T [I(W(x, y; \mathbf{p})) - T(x, y)], \tag{3.19}$$

(compare to (3.11)), and the Hessian matrix is the same one as in (3.12) with the roles of I and T changed:

$$H = \sum_x [\nabla T \frac{\partial W}{\partial p}]^T [\nabla T \frac{\partial W}{\partial p}]. \tag{3.20}$$

As noted before, the Jacobian is fixed and computed at $(x, y; \mathbf{0})$. In addition the gradient ∇T is evaluated at $W(x, y; \mathbf{0})$. Therefore the Hessian H is not dependent on $\mathbf{p}$ and can be pre-computed and inverted once. At each iteration, only the following term is computed:

$$\sum_x [\nabla T \frac{\partial W}{\partial \mathbf{p}}]^T [I(W(x, y; \mathbf{p})) - T(x, y)] \tag{3.21}$$

and the warp parameters are updated according to (3.16).

In [5], the equivalence between the inverse compositional algorithm and the forward compositional algorithm is proved. In addition, the algorithm works best under the assumption that the change to the parameters $\Delta \mathbf{p}$ is small. This algorithm can be used to speed up our alignment procedure in a very similar way by changing the roles of the image and the template.

3.2.3 Feature-based alignment

A different approach to alignment is a feature-based method based on extracting visual features in each of the images and matching them. Such methods find local features that are preferably invariant to translation, scaling and rotation and are distinctive. Therefore, distinctive features are detected in two images. In some applications, we have prior information about our images, so the feature detection can be improved (for example, in images of a rugby field, we might be interested in lines indicating the distance from the goal-line). The coordinates of the matched features are used to com-

pute the homography. Matching can be challenging as well, since we will usually not match all features, but only the strongest ones. However, we would not like to use outliers for matching, but choose only the best features. A simple approach will use RANSAC to iteratively pick up a number of matches and compute some measure of merit.

In the general case of two images without prior knowledge, a commonly used set of features is Harris corners [40]. A corner is defined as the location of a pixel which has a strong edge response in more than one direction. A corner is detected by centering a small window over a pixel and then shifting the window slightly both horizontally and vertically. The sum of differences of intensities of the pixels in the original window and the shifted window should be high in any direction the window is shifted. An edge does not satisfy this requirement as there is no change along the edge direction. A simple alignment algorithm will detect the strongest Harris corners in both images and match these coordinates. The matching is performed by looking at each Harris corner and matching its coordinates with a corner with similar dominant direction of the gradient in a nearby spatial location.

Another very common set of features is SIFT [80]. The approach produces a feature vector which is invariant to scale and rotation. First, keypoints are detected using a scale-space approach. At each scale, a set of images is produced by using the Difference of Gaussian (DOG) operator with varying σ. The transition between scales is done by convolving the original image with 2σ and resampling the image. For each image in every scale, keypoints that are local minima or maxima are taken. They are detected by comparing each pixel in the DOG image to its immediate neighbors in the same scale and the two consecutive scales (26 neighbors). However, a lot of the detected pixels are edges, so a similar computation used to detect Harris corners is employed. Only pixels that have a small ratio between the eigenvalues of their Hessian matrix are kept. A high ratio usually means that the pixel has a dominant direction, thus probably an edge.

The feature vector is built by examining a region centered at the keypoint. The algorithm looks at a window of size 16×16, and divides it to 16 sub-windows of size 4×4. For each sub-window, an histogram of 8 bins is built, where each bin is

related to the orientation of the gradients. A pixel is associated with a bin according to its gradient's orientation. The contribution of each pixel to the bin is a function of its gradient's magnitude, the distance of the pixel from the window's center and the current scale. The feature vector is built of the orientation histogram entries and is of size: $8 \times 16 = 128$. This vector is invariant to scale and rotation and it has been found that smaller feature vectors do not perform so well.

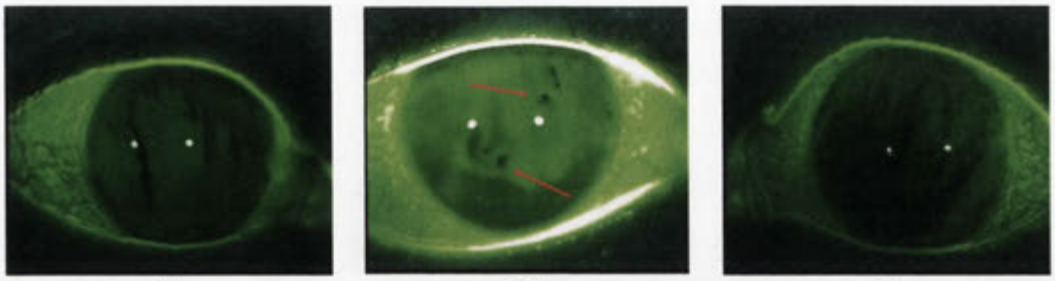
The SIFT features are computed on both images and a feature matching technique is employed to find the best homography. The closeness of two feature vectors can be defined by their Euclidean-distance. Approaches for matching include k-nearest neighbor techniques, k-d trees, RANSAC and more. When having multiple images, such as in a video, the features can be tracked over multiple frames to find the homography between all images in the video. Such approach is suitable even when the video has large changes between image frames.

3.3 Non-automatic methods to assess the tear film quality

In this section, I will discuss very briefly other methods to evaluate the tear film quality in images of the anterior of the eye. While there is an extensive work to build dedicated hardware to analyze the tear film or to manually collect clinical data, very few methods are automatic that do not require manual workload by a clinician.

In an interesting work by **Bitton and Lovasik** [10] that is clinically related to ours, the authors try to correlate the patterns of the rupture of the tear film to the symptoms of dry eyes. They note that the BUT is a single measurement that does not offer any qualitative or quantitative information about the form of the break. Therefore, analyzing the tear film as it changes over time as opposed to the single snapshot that reflects the BUT, can provide further information regarding the changes of the tear film between blinks. They perform the FBUT test and record the video. This video is digitized and analyzed by using an image editing software.

A clinician analyzes the video for the frame where the rupture of the tear film is fully developed and records this image. Then the following attributes are measured: the BUT, the location of the break using the CCLRU standards , the shape of the break,



(a) (b) (c)

Figure 3.4: *The shapes of rupture as defined by Bitton and Lovasik. (a) Streaks (b) Dots (c) Pools*

the evolution rate of the break and the interval between the FBUT and the next blink. The possible shapes for the rupture pattern are: streaks, dots and pools. The authors assume that there is only one major break that has to be analyzed in every sequence. An example for the different rupture patterns is shown in Fig. 3.4.

In their analysis, the authors try to find a correlation between: the pattern of the break and the region it appears; the BUT and the break pattern; the time between the break and next blink (TBUT-to-Blink) and the break pattern; the total blink interval and the break pattern; and analyze the progress of the area of the break. Their results demonstrate that streak-line patterns are significantly different from dot-like patterns in terms of the BUT, TBUT-to-Blink and the time it takes the rupture to evolve.

The authors could not come to a conclusion regarding the repeatability of the break patterns, since they used a small sample of patients. More experiments are needed to improve the understanding of the tear film.

The main reason for bringing up this paper is to demonstrate the potential in further analyzing the rupture patterns when using the FBUT test. The manual collection of such data and analyzing it by a clinician is a very time-consuming task. Proving repeatability requires recording hundreds of videos of patients with dryness symptoms, which are acquired over a period of time. In chapter 5, we show how all these symptoms are analyzed in an automatic way without the intervention of a clinician.

Begley and el. [7] analyze the size of the break in relation to the BUT and the total blink to blink interval. They performed the FBUT test on control and dry eye subjects and repeated the measurement three times. The BUT is detected manually by a clin-

ician. The area of the break is highlighted, assuming that the break areas are darker than the relative fluorescence at that location. Finally, the size of the highlighted area is detected automatically by a software.

Their results show a strong correlation between the size of the break to the blink to blink interval and the BUT. Bigger break areas correspond to shorter blink to blink interval and BUT. In terms of the size of the break, control subjects tended to have discreet areas of break evolving slowly. Dry eye subjects showed rapid disruption of the tear film, with dry areas growing rapidly in size. These areas are on average bigger than those of the control subjects.

The method can be used in clinical trials to add quantitative data regarding the break size in addition to the BUT that is already measured by clinicians. However, the detection of the region of the break area and size is not done automatically. In the paper, it is assumed that the fluorescein distributes uniformly over the corneal area. An attempt to automate the detection of the dry areas in the image using this assumption will usually not work. In our experiments, we have found that different light conditions and uneven spreading of the fluorescein happen from time to time. Using a prior image of the relative fluorescence over the iris and looking at a difference image between the last image before the blink and the prior tends to fail. However, if we can create a reliable method to measure the amount of fluorescein throughout the video, it can be very useful in cases of uneven distribution or when it keeps spreading throughout the video. However, this is not an easy task.

Goto and Tseng [35] use kinetic analysis to investigate how the pre-corneal lipid film spreads and distributes. They record a video focusing on a certain area of the cornea using a dedicated device based on a biomicroscope (without instilling fluorescein). The video shows the speed and direction of spreading of the tear film between blinks. The first video frame where there is no noticeable movement of the film is defined as the first stable image. This image is used to analyze the thickness of the lipid layer.

They discovered that the lipid film spreads vertically upward from the lower lid after the blink. The average spreading time for dry eye subjects was 6 times longer on average. Because of the slower spread, the resulting lipid film was found to be thicker

on the inferior part than the superior part of the cornea for dry eye subjects. For non dry-eye subjects, the lipid layer thickness did not vary much between the inferior and superior parts. The results were computed using 17 dry-eye patients so a much bigger sample is needed before a conclusion can be made. Furthermore, the manual task of scanning every video frame for changes is a very tiring one.

To conclude, we showed a few recent non-automatic methods to evaluate the quality of the tear film. The methods use imaging of the anterior of the eye, but vary in the symptoms that are evaluated. The main difficulties are the manual labor required by the clinician, the added subjectiveness of any manual result and the need for a big sample to prove the clinical usability of a new device.

Chapter 4

Markov Random Fields and Graph-Cuts

This chapter provides a short introduction to the **graph-cut** algorithm and its usage for binary segmentation of an image. The binary graph-cut algorithm finds the optimal partition of a graph to object and background (for a given cost function). We use this algorithm later in the thesis in chapters 6 and 8.

4.1 The labeling problem

We define the labelling problem in terms of random variables and labels. We denote the set of variables as:

$$\mathcal{X} = \{X_1, \dots X_n\} \tag{4.1}$$

and the set of labels as:

$$\mathcal{L} = \{l_1, \dots, l_m\}. \tag{4.2}$$

Each variable from $\mathcal{X}$ can be assigned any label from $\mathcal{L}$. Let $\mathbf{x} = \{x_1, \dots, x_n\}$,$x_i \in \mathcal{L}$ be a set of of variables. We denote by x_i the assignment of a label to random variable i. We are looking for a valid labelling that assigns a label from $\mathcal{L}$ to each random variable in $\mathcal{X}$.

In binary segmentation applications, we would like to segment the image into object and background. In that case, each pixel in the image is a random variable in

$\mathcal{X}$ and the label set is:

$$\mathcal{L} = \{1, 0\}, \text{ or } \mathcal{L} = \{object, background\}. \tag{4.3}$$

Each valid labelling assigns one of the two labels to all pixels, therefore segmenting the image into two individual objects. Computing the probability of such labelling is not easy, as images hold contextual constraints. Therefore, the variables are not mutually independent.

4.2 Markov Random Fields

The variables in $\mathcal{X}$ are related to each other via a neighborhood structure $\mathcal{N}$. For each variable $X_i \in \mathcal{X}$, a neighborhood $\mathcal{N}_i$ is defined as a subset of $\mathcal{X}$. For example, in a 2D lattice (image), we can use a 4-connected or a 8-connected neighborhood. Then variable X_i will have 4 or 8 neighbors respectively as shown in Fig. 4.1(a). The definition of a neighborhood states that a variable cannot be a neighbor of iteself, and that the relationship is mutual (if $\mathcal{X}_j \in \mathcal{N}_i$ then $\mathcal{X}_i \in \mathcal{N}_j$). F is said to be a Markov Random Field (MRF) on $\mathcal{X}$ with respect to a neighborhood system $\mathcal{N}$ if the probability of every labelling is bigger than 0 and the conditional probability distribution of each random variable $\mathcal{X}_i$ is only dependent on its neighboring variables $\mathcal{N}_i$. [14, 96].

For example, given an image I of size $N \times N$, the location of each pixel is defined using two indices (x, y), such that its intensity is given by $I(x, y)$. This can be converted easily to an MRF:

1. A random variable is assigned for each pixel. The pixels can be enumerated in the order of scanning the image using two indices. Then the set of random variables is:
   $\mathcal{X} = \{X_{1,1}, X_{1,2}, \ldots X_{1,N}, X_{2,1}, \ldots, X_{2,N}, \ldots, X_{N,N-1}, X_{N,N}\}$.

2. The neighborhood $\mathcal{N}$ of pixel (x, y) can be defined as a 4-connected structure: $N_i = \{(x-1, y), (x+1, y), (x, y-1), (x, y+1)\}$. Alternatively, it can be seen as $N_i = \{left, right, up, down\}$. In a 8-connected neighborhood, each pixel will

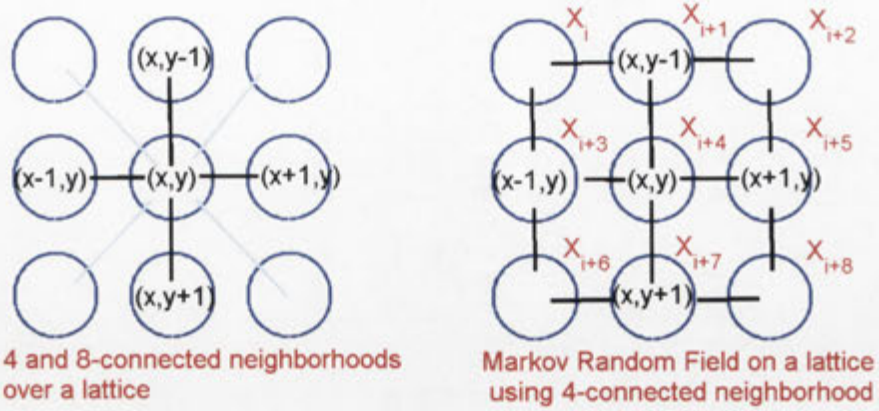


Figure 4.1: *(a) Neighborhoods on a lattice. The 4-connected one is plotted using the black lines. The 8-connected one includes both the black and turquoise lines. (b) An example of MRF over a lattice using a 4-connected neighborhood. The X_{ij}'s correspond to the random variables (or pixels). For example, pixels $X_{i,j}$ and $X_{i+1,j}$ are mutual neighbors.*

have 8 immediate neighbors. A conditional probability is assigned to each variable $\mathcal{X}_i$, such that: $P(X_i = x_i | X_j = x_j; j \neq i) = P(X_i = x_i | X_j = x_j; j \in N_i)$.

Therefore, the random object $\mathcal{X}$ is said to be an MRF on the lattice (x, y) with neighborhood system $\mathcal{N}$. Figure 4.1(b) depicts an MRF over a lattice using a 4-connected neighborhood. Each random variable X_{ij} has 4 neighbors.

4.3 Gibbs distribution

A clique c over a graph $G = (V, E)$ is defined as a subset of the vertices V, such that all pixels in this subset are connected to each other by an edge. Thus, for all vertices v_i and v_j in a clique c, there is an edge $(v_i, v_j) \in E$. Note that a path between v_i to v_j is not enough for the vertices to be in c, but a direct edge is required. Using the definitions from last section, the notion can be extended to a lattice (image). When using a 4-connected neighborhood (see Fig. 4.2), the cliques are : (1) the pixel itself (unary cliques): $C_1 = \{X_i\}$ (2) two neighboring pixels (pairwise cliques): $C_2 = \{X_i, X_j\}$. Similarly, in a 8-connected neighborhood the maximum clique size is also 2, as the neighborhood structure can be seen as 8 individual pairs. We define the set $\mathcal{C}$ to be all

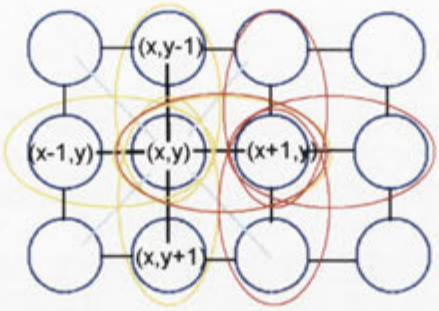


Figure 4.2: *Cliques on a lattice. The figure is similar to Fig. 4.1(a) with a larger lattice. The cliques of size two that include pixel (x, y) are plotted in yellow when using a 4-connected neighborhood. However, one can see that even when considering the 8-connected neighborhood, the maximum clique size is still two. Similarly, all cliques that include pixel $(x + 1, y)$, plotted in red, are also of size two.*

the cliques in a graph.

A set of random variables $\mathcal{X}$ is said to be a Gibbs distribution [34] with respect to a neighborhood $\mathcal{N}$ if the joint distribution takes the following form:

$$P(X = \mathbf{x}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} exp(-E_c(\mathbf{x})). \tag{4.4}$$

In this equation $\mathbf{x}$ is a labelling assigned to the random variables in $\mathcal{X}$; The term Z is a normalizing constant; E_c is some energy function that depends on the values of the random variables of $X_i \in c$. In other words, for a given clique $c \in \mathcal{C}$ the energy function E_c is only dependant on the random variables that belong to that clique. The whole equation states that the probability of a labelling $\mathbf{x}$ is defined as the product of energy functions E_c, where $c \in \mathcal{C}$ is any clique defined in $\mathcal{X}$. However, each energy function E_c is only dependent on the variables in the clique c and not over all possible combinations between variables (which would be a very large number even for small sets $\mathcal{X}$).

The Hammerseley-Clifford theorem [34, 76] relates an MRF to the Gibbs Distribution: $\mathcal{X}$ is an MRF with respect to $\mathcal{N}$ if and only if the set of random variables $\mathcal{X}$ is a Gibbs Distribution with respect to $\mathcal{N}$. This is a very powerful theorem when used on lattices. In previous section, we showed how to define an MRF on a lattice with a maximum clique size of 2. Using the Hammerseley-Clifford theorem, we infer that any labelling $\mathbf{x}$ is computed as a product of energy functions, each depends on a max-

imum of 2 random variables (two pixels). Therefore, the MRF presented in 4.1(b) is also a Gibbs Distribution with unary and pairwise cliques. The labelling problem is now defined as finding an assignment **x** which maximizes the probability (4.4). Taking the log of (4.4) returns:

$$\log P(X = \mathbf{x}) = const - \sum_{c \in \mathcal{C}} E_c(\mathbf{x}). \tag{4.5}$$

Hence,

$$argmax_{\mathbf{x}} P(X = \mathbf{x}) = argmin_{\mathbf{x}} \sum_{c \in \mathcal{C}} E_c(\mathbf{x}). \tag{4.6}$$

Thus, the best labelling can be found by minimizing the energy function:

$$\sum_{c \in \mathcal{C}} E_c(\mathbf{x}). \tag{4.7}$$

In the special case of cliques of size up to two, for example, when using a 4-connected neighborhood, (4.7) can be simplified. We denote E_i as a cost function that depends only on one variable i, and E_{ij} as a cost function that depends on two variables i and j, the labelling problem can be written as finding the minimum of:

$$E(\mathbf{x}) = \sum_{i \in \mathcal{X}} E_i(x_i) + \sum_{(i,j) \in \mathcal{N}} E_{ij}(x_i, x_j). \tag{4.8}$$

4.4 Pseudo-Boolean functions

Most of the material in this chapter can be found in [13], using slightly different notations. A pseudo-boolean function is a mapping from $\mathcal{B}^n$, where $\mathcal{B} = \{0, 1\}$ to the real numbers:

$$f : \mathcal{B}^n \rightarrow \mathcal{R}. \tag{4.9}$$

The function f has n variables, where each one can have a label of 0 or 1. Therefore, a real number is assigned to every combination of the n variables. The most simple way to represent a pseudo-boolean function is by listing all options in a table with the corresponding costs as in Table 4.1. The function f is defined as the sum of all terms.

x_1	x_2	value	term
0	0	3	$\bar{x}_1\bar{x}_2$
0	1	-2	$\bar{x}_1 x_2$
1	0	4	$x_1\bar{x}_2$
1	1	-5	$x_1 x_2$

Table 4.1: A function of two variables and the corresponding posiform terms

For example, in Table 4.1, the function f is:

$$f(x) = 3\bar{x}_1\bar{x}_2 - 2\bar{x}_1 x_2 + 4x_1\bar{x}_2 - 5x_1 x_2. \tag{4.10}$$

Therefore, for a given assignment all terms but one will equal zero. For example, there are 4 ways to assign values to two variables x_1 and x_2, as listed in Table 4.1. However, for every given assignment, 3 out of the 4 terms in f will equal to zero. It is clear that the minimum of the function is achieved when $x_1 = 1$ and $x_2 = 1$ as the function's value is -5.

We define a posiform as:

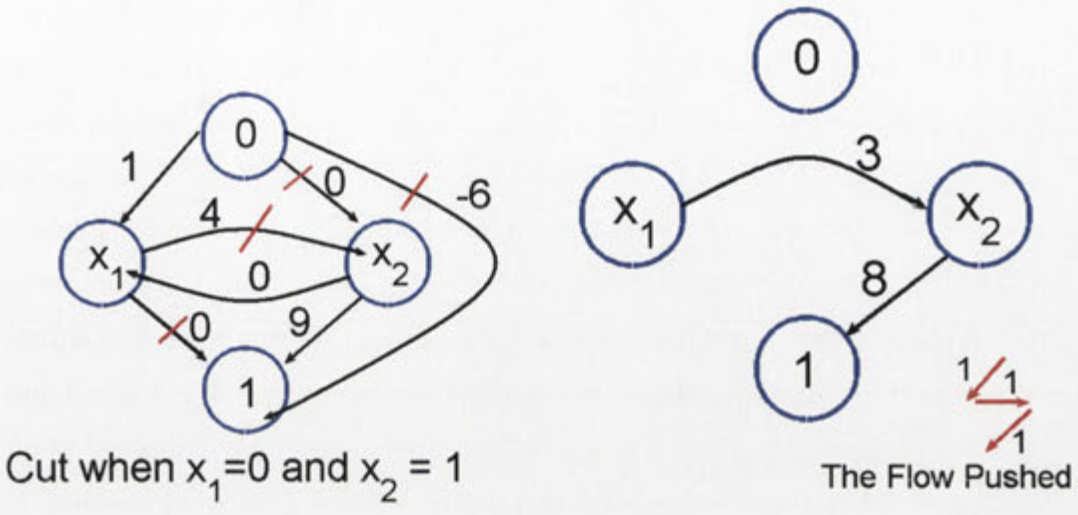
$$f(x) = a_0 + \sum_i a_i u_i + \sum_{i,j} a_{ij} u_i u_j + \ldots, \tag{4.11}$$

where u_i and u_j are literals (such as x_i and $\bar{x}_i$), a_0 is a constant and all coefficients (a_i and a_{ij}) are positive. If all assignments to all variables of the given function f are positive($a_i > 0$ and $a_{ij} > 0$), then the sum of the terms is a posiform. However, the function of (4.10) is not a posiform, since two assignments are negative. It can be adjusted to a posiform by changing literals to their complements to receive a posiform:

$$-5 + 5\bar{x}_2 + 3\bar{x}_1\bar{x}_2 + 3\bar{x}_1 x_2 + 4x_1\bar{x}_2 \tag{4.12}$$

This term is a posiform, since all coefficients are positive except for the constant. Always all negative terms can be changed to positive in a similar way.

It can be shown that a quadratic cost function given by (4.8) can always be converted to a posiform as in (4.11). Therefore, we can find the optimal labelling of (4.8) by finding the minimum of the equivalent posiform.



(a) (b)

Figure 4.3: *(a) A graph of two variables. The cut here is marked by the red lines and its cost is -2. This function is submodular since all weights between pixels are non-negative (the negative weight corresponds to the const a_0). (b) The graph after pushing a flow of 1 (and removing the edge from 0 to 1). Edges with a zero weight are removed. The maximum flow is 1 (as no more flow can be pushed between 0 and 1), and the minimum of the function is achieved when $x_1 = 1$ and $x_2 = 1$, as seen by the partition of the graph*

4.5 The graph-cut algorithm

First we show how to build a graph G from a given pseudo-boolean function $f(\mathbf{x}) = f(x_1, \ldots, x_n)$. We create a vertex for each variable and add two more vertices called terminals and denote them by 0 and 1. Therefore, in total we have $n + 2$ vertices. In addition there are certain edges with weights. A **partition** or a **cut** is defined as a division of the graph to two sets $\mathcal{V}_0$ and $\mathcal{V}_1$. The vertex 0 always belongs to $\mathcal{V}_0$ and the vertex 1 to $\mathcal{V}_1$. For a given labelling $\mathbf{x}$, each variable can have a value of 1 or 0. According to their value, we associate the corresponding vertices to exactly one of the two partition sets. We define the **cost** of such partition as the sum of the weights of all edges going from $\mathcal{V}_0$ to $\mathcal{V}_1$.

A function $f(\mathbf{x})$ is said to be **simple graph representable** if there is a graph, such that every assignment $\mathbf{x}$ of the function $f(\mathbf{x})$ equals the cost of the partition (cut). For example, assume the function in Table 4.1 and a specific labelling: $x_1 = 0$ and $x_2 = 1$.

Therefore x_1 belongs to the set $\mathcal{V}_0$ and x_2 to the set $\mathcal{V}_1$. The aim is to show that it is possible to assign edge weights, such that when removing all edges connecting $\mathcal{V}_0$ and $\mathcal{V}_1$, their sum equals $f(0,1) = -2$. Fig. 4.3(a), shows such a labelling. There are 4 edges from 0 to 1 and by removing them we create a cut. The sum of these edges is -2 as required. We observe that the other 3 values associated with the labellings in Table 4.1 also equal to their corresponding cuts in 4.3(a). Thus, it is a simple graph representation of the function. Therefore, the cut with the minimum cost equals the minimum of the function. This is achieved when assigning $x_1 = 1$ and $x_2 = 1$ and the cut equals $1 + (-6) = -5$. Further details on how to build the graph and to fill the weights of the edges can be found in [13]. It can be shown that it is possible to build such graph for every posiform in the format of (4.11).

The algorithm which separates the graph into two sets is called the **min-cut** algorithm and is also known as the **graph-cut** algorithm. It has been demonstrated that the solution to the graph-cut algorithm equals the **max-flow** on the same graph. Max-flow can be computed efficiently using the Ford-Fulkerson algorithm [26].

A flow on a graph can be thought as pushing water through the edges of the graph, where each edge has a capacity w_{ij} which equals its weight. Every amount of water that is pushed in has to also get out. A negative flow from vertex u to v means that the actual direction of the flow is from v to u. A **permissible flow** also requires that the amount pushed from vertex u to v is not higher than the edge capacity w_{ij}. Therefore, in order to have a permissible flow on a graph, the sum of weights between any two vertices must not be negative: $w_{uv} + w_{vu} \geq 0$. Fig. 4.3(b) demonstrates the resulting graph after pushing a flow of 1 in the graph of fig. 4.3(a).

The requirement of having a permissible flow can also be written in terms of the costs used in (4.8). A quadratic function of the form of (4.8) is **submodular**, also called **regular**, if it satisfies the following condition:

$$E_{ij}(0,1) + E_{ij}(1,0) - E_{ij}(0,0) - E_{ij}(1,1) \geq 0. \tag{4.13}$$

This representation of submodularity also appears in [65], but is derived in a different way than the one shown here, based on re-parametrization of the graph. The require-

ment in (4.13) states that the cost of assigning different labels to neighboring pixels should be higher or equal to the cost of assigning the same label.

The immediate consequence is that in order to solve the graph-cut problem, the graph representation of the function needs to have a permissible flow. In other words, the sum of weights between all vertices in the graph cannot be negative. Therefore, given an input image, a neighborhood with a maximum clique size of 2 (such as 4-connected neighborhood) and a binary label set such as in (4.3), the cost function can be converted to its simple graph representation form. A solution using the graph-cut algorithm can be found if and only if the function is submodular. We note that some non-submodular functions can also be minimized using different techniques such as the roof-dual algorithm presented in [13].

4.6 Graph-cuts extensions

The graph-cut algorithm has been used extensively in recent literature. This is mainly since the global optimum of many cost functions can be found in polynomial time. Some of the usages of graph-cuts include: segmentation [61, 70, 97], including medical images and 3D volumes [3, 15, 16, 36, 56, 119], stereo reconstruction [64, 123] and image restoration [114].

An extension to the original algorithm to segment a number of objects is given in [17]. The paper presents two algorithms named **alpha-beta swap** and **alpha-expansion**. The output of these algorithms is a segmentation of the image into a number of different labels (objects). Therefore, instead of running the binary graph-cut algorithm a few times on the image to segment multiple objects, these algorithms use the relationship between objects to produce simultaneously a multi-object segmentation. The alpha-expansion is a relatively fast algorithm, but is not guaranteed to produce the global optimum. However, in practice, the results have been shown to be very close to the optimum. I will further discuss the alpha-expansion algorithm in section 6.2.

Kohli and Torr present a way to reuse the graph-cut solution when a similar graph-cut problem has to be solved afterwards. They call their method dynamic graph-cuts [62]. In some labelling problems, we would like to change some of the parameters

of the energy function to achieve a lower cost. Thus, the unary and pairwise terms can change for some nodes, but many nodes still retain their original cost. Instead of solving the problem again with the new parameters, the graph and the max-flow solution from the previous iteration can be reused to accelerate the computation.

Their solution first runs the max-flow algorithm on the original graph. Then in subsequent iterations, the solution uses the max flow obtained in the previous iteration and further flow is pushed through the graph to obtain a new solution. Such operations can change the maximum permitted flow, but do not change the min-cut. Therefore, the labelling does not change as well and a more efficient solution can be achieved by reusing the previous flow.

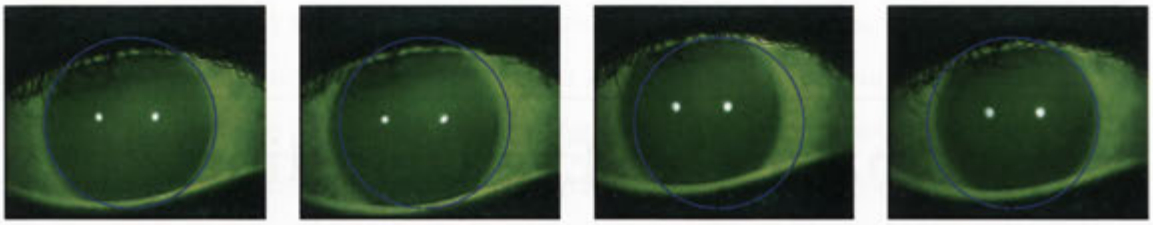
Chapter 5

Automatic Dry Eye Detection

The Fluorescein Break Up Time Test (FBUT) (see chapter 2.3.2) is the test of tear film stability most commonly used by clinicians as it is minimally invasive [18]. A few improvements have been presented over the years including a method to control the amount of fluorescein instilled [69]. It has already been pointed out in [90] and [10] that using this test alone is not enough. Still in most cases, this is the only measure used by clinicians to diagnose dry eyes and to decide what treatment to give, even though the test has a few objective limitations:

1. High subjectiveness - The decision when the break happens is highly subjective and is dependent on the clinician.
2. No option to review - The clinician usually have only one opportunity to measure the FBUT .
3. Reproducibility - The FBUT test is not reproducible in terms of the amount of fluorescein instilled and illumination.
4. Inconsistency - Significant inter-observer and intra-observer differences in the measured FBUT between the qualified clinicians. This is different than the reproducibility limitation, since the inter-observer variance is defined on the same videos using several clinicians (see subsection 7.1.1).
5. Record - No record of the examination is kept.

The main contribution of this chapter is the development of new methods and ideas to automatically measure a range of symptoms related to dry eyes. We have defined a new value called Digital Electronic Break Up Time (DEBUT), which is clinically



(a) (b) (c) (d)

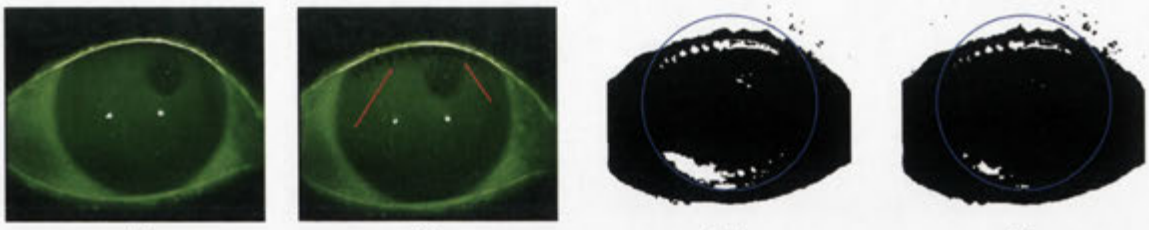
Figure 5.1: *Demonstrating the movement of the iris in a sequence even when the camera is mounted. In each of the images, the iris is drawn at the same location as it is in the first image.*

equivalent to the BUT, but is defined in different ways based on temporal image properties. This value is not operator-dependent, reproducible and robust to illumination changes. It also enables one to carry out multi-center studies.

Our method creates a **dryness image** which depicts the different degrees of tear film thinning over the corneal area. Each connected area in the dryness image is analyzed to include its size, shape, location and progress. Dry areas near the upper and lower menisci are detected separately as they are clinically related to dryness originating from the tear reservoir. Bitton and Lovasik [10] investigate some of these cues over a sample of patients (see chapter 3.3). They show the importance of analyzing the rupture patterns of the tear film in conjunction with the BUT , but do not offer any automatic measures to do so. We provide this information as part of the analysis, but more importantly, it enables us to correlate the rupture patterns over a wide range of patients with the symptoms in an automatic way.

5.1 Demonstrating the difficulty

Before discussing our approach, I would like to demonstrate a few of the challenges any method that tries to detect the dry areas and the BUT faces. The main difficulty is also related to the most important feature of the camera: it is small and hand-held (see Section 2.3.3). It is very difficult to hold the camera still, such that the iris is located at the same location throughout the video. Furthermore, even in experiments where the camera was mounted on a stand, there was a significant movement of the iris, as the patient's gaze cannot always stay still for long periods. Fig. 5.1 shows 4 images

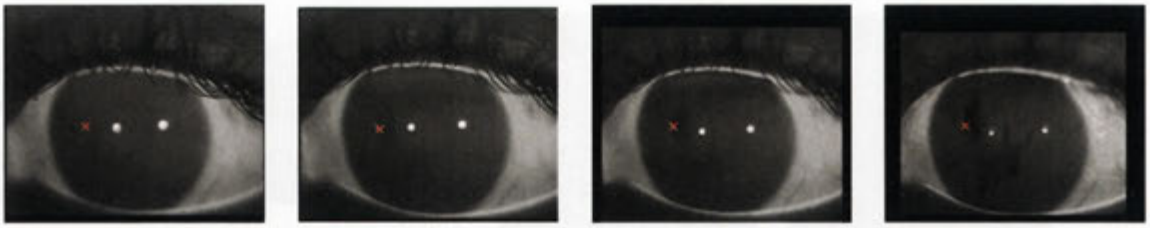


(a) (b) (c) (d)

Figure 5.2: *Demonstrating the challenge in segmenting the dry areas when the images are unevenly illuminated. (a) Image after the blink. (b) Image before the next blink. The arrows point to regions of dryness. (c) The image in (b) using a threshold of 50. (d) The image in (b) using a threshold of 47.*

taken from a video sequence where the camera was mounted. The frame numbers are 0, 110, 262 and 400 - the last frame before the blink (25 frames per second). In the first image, the iris is detected and highlighted. In the rest of the frames, the iris is drawn at the same location as in the first frame. It is evident that the iris keeps moving throughout the video. When the camera is hand-held, the movement of the iris can sometimes span half of the image (depending on the image resolution and the iris magnification). Therefore, any algorithm that detects the dry areas and their progress needs to be able to track the iris over time.

The second challenge is related to the way the images are illuminated. If the patient looks straight into the camera, the camera is held still, the room has no external light sources and the fluorescein spreads evenly, the iris should be illuminated evenly. However, this is rarely the case. Quite often the camera is angled in relation to the iris, resulting in uneven illumination. The immediate consequence is that dark areas do not always relate to dryness, and even if they do, it does not necessarily show the **degree** of dryness. Fig. 5.2 demonstrates such a video: part (a) is the image immediately after the blink. One can see that the inferior (lower) region is darker than the superior (upper) region. Part (b) depicts the image before the next blink. The arrows point to regions of break. The images have been converted to gray-scale using intensities from 1 to 255. Parts (c) & (d) of the figure show the last image after using a threshold of 50 and 47 respectively and the iris is highlighted. In part (c), big chunks of non-break areas are included in the inferior part (possibly there is some degree of thinning there). In part (d), most of the dry areas are segmented, but some small break areas



(a) (b) (c) (d)

Figure 5.3: *Demonstrating the difficulty in detecting the exact timing of the BUT. (a) First image after a blink. Images after aligning the iris at times: (b) 1.5 seconds (c) 3 seconds (d) 7 seconds. The images at these times correspond to different values of BUT given by clinicians.*

in the superior region are already not included. Small regions are also incorrectly segmented in the inferior part. This example demonstrates that sometimes even the optimal threshold cannot segment these areas. Furthermore, only a slight increase of the threshold (from 47 to 50) results in a wrong segmentation.

Finally, I will demonstrate one (among many) challenges in detecting the accurate time of the break. Clinically, the BUT is defined as the time when dark spots appear. Fig. 5.3 shows 4 images taken from a video where the iris has been aligned (so the iris appears at the same location and size) at times: 0, 1.5 seconds, 3 seconds and 7 seconds. The length of the sequence is in total 14 seconds. Looking at the images it is clear that some areas are becoming darker and there is a break. For example, the pixel marked with an 'x' has the following intensity in the four images: 66, 52, 42 and 35. Clearly this pixel shows steady decrease in intensity and increase in the degree of the break. However, is the BUT occurring at the time of the second image, third image or the fourth one? To add to the confusion, these images are related to actual break up times (BUT) given by three different clinicians! The difficulty in this sequence can be described as how one can objectively estimate blackness. In addition, the following question arises: is the BUT defined as the time when an area **starts** to get darker or as the time it **does not become any darker**?.

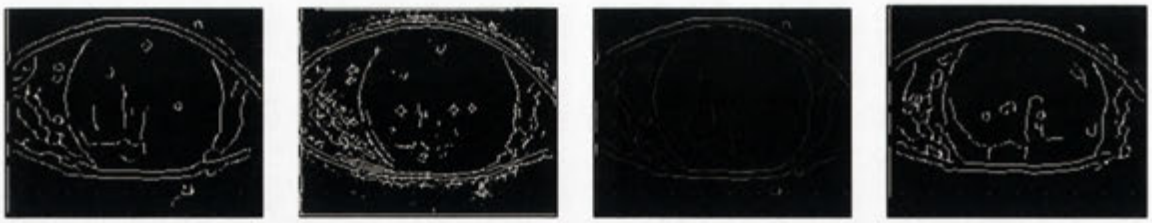
5.2 The algorithm in brief

The dry areas always appear as darker areas on the iris in the fluorescein image when compared to the same spot in previous images. As demonstrated in the previous section, the method needs to handle the movement of the iris and find an objective value for the BUT . Given the recorded video, the algorithm is based on three main steps:

- Accurate detection of the iris and eyelids in the first frame of the video. Then the iris is detected in the rest of the images using the location of the iris in the previous image and a fast minimization method.
- Alignment of the images - The aim is to create a video where the iris is at the same location and size throughout the video. Using the estimated location of the iris in each image and the grey level values, the iris is aligned to the iris in the previous images. In the output video, dry spots should appear at the same location throughout the video.
- Segmentation and analysis of the break area and computation of the break up time. Images from the beginning and end of the aligned video are analyzed to create dryness images of the dry areas and regions near the tear reservoirs. Then, the whole video is scanned to find the accurate timing of the BUT and to analyze the progress of individual break areas.

5.3 Detection of the iris and eyelids

The images of interest in the video are those between two consecutive blinks. All captured videos are of resolution 352×288. We call such a set of images a **sequence**. To that end, we first find all the blinks and incomplete blinks and treat each sequence individually. A **sequence start** is defined as the frame where the upper lid upward motion stops. A **sequence end** is defined as the frame where the upper lid downward motion starts. Blinks are usually easy to detect, as a few consecutive images will have big differences in intensity. An **incomplete blink** happens when the upper lid's



(a) (b) (c) (d)

Figure 5.4: *(a) Threshold image of the first frame in the sequence (a) using Canny. (b) using Sobel. (c) Magnitude image of (a). Notice that the strongest gradients belong to the eyelids and noise and not necessarily to the iris. (d) Last frame in the sequence using Canny. The noise is quite noticeable especially at the last frame. Also notice the double edges for the eyelids.*

downward motion stops before covering the whole iris area. They are usually harder to detect as they last only for a few frames. After an incomplete blink, it is still of interest to measure the time passed until a full blink happens, even though dryness in some areas is less relevant.

Given an image $I(x, y)$ after a blink, we create an edge map $E(x, y)$ of the image using the Canny edge detector. The value of each pixel is the magnitude of the gradient. We found that Canny produces the best results, mainly due to the smoothing which connects the edges and the removal of noisy disconnected areas as seen in Figs. 5.4(a&d). The parameters for the Canny have been learnt from using a set of videos, but are adjusted at runtime. See the comment at the end of this section. The superiority of the Canny over the Sobel edge detector is shown in Fig. 5.4(b). The Sobel image includes much more noise and less smoothing.

We create 3 threshold images of the iris, the lower eyelids and the upper eyelids using 3 different thresholds:

$$\begin{aligned} I_{\text{iris}}(x, y) &= (E(x, y) > T_1) \\ I_{\text{low}}(x, y) &= (E(x, y) > T_2) \\ I_{\text{up}}(x, y) &= (E(x, y) > T_3) \end{aligned} \tag{5.1}$$

We use $T_1 \leq T_2 \leq T_3$ as the edges of the eyelids are usually stronger than the border of the iris, which is fuzzy due to the fluorescein spreading. The eyelids tend to have

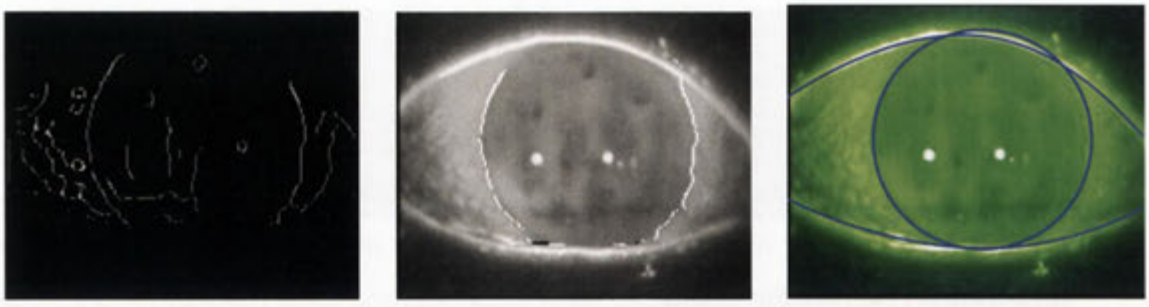
strong bright edges due to the fluid found in them. Usually, the upper eyelids will have stronger edges than the lower eyelids. This is clearly visible in the magnitude image in Fig. 5.4(c). The brightest pixels are the strongest gradients and they mostly belong to the eyelids and not to the iris. Thus, performing a direct fitting to the iris is impractical due to the high ratio between the inliers (the iris pixels) and the outliers (everything else), see Fig. 5.4(a).

To overcome the aforementioned problems, we first fit a polynomial of degree 2 to the eyelids. In most cases, it is an adequate fitting, as we are mainly interested in the iris area. The eyelids can curve more in the conjunctival area where a polynomial of degree 2 is not enough. To that end, we use RANSAC [41] with three parameters (a,b,c) to fit a polynomial $ax^2 + bx + c$ to I_{up} and I_{low}. As we expect the upper eyelids to be concave and the lower to be convex and the curvature to be small we require that $a > 0$ and $b < 0$ for upper eyelids and $a < 0$ and $b > 0$ for lower eyelids and also $0.001 < |a| < 0.01$. At each iteration, three points are sampled and a vector $R(i)$ of residuals is returned, where i is a pixel in the threshold edge image. If a pixel i falls on the estimated curve, we assign $R(i) = 1$ otherwise $R(i) = 0$. The confidence value returned by the RANSAC is therefore the sum of the values in the vector R.

However, as depicted in Fig. 5.4(d), a few curves for the eyelids can exist. We would like to make a fitting to the closest curve to the iris in order to avoid including the high intensity pixels of the eyelids. To that end, for a given estimation of the eyelids, the objective function searches whether a similarly shaped curve (or only a partial curve) exists above or below it. For example, for the upper eyelids, we penalize the objective function if a curve is detected below it. We credit the objective function if a curve is detected above it. Therefore, the lowest closest curve to the eyelids should score higher. Similar arguments hold for I_{low} as well.

The pixels above and below the upper and lower eyelids respectively in I_{iris} are removed. The resulting image is shown in Fig. 5.5(a). It can be seen that the iris is more prominent in the image, although a fair amount of noise still exists.

In order to fit a circle to the remaining pixels, we use RANSAC with three parameters (x, y, r) and sample three points at each iteration. Similar ideas to the eyelids fitting are used, imposing restrictions over the points distribution and limiting the ra-



(a) (b) (c)

Figure 5.5: *Results for fitting the iris and eyelids. (a) The image of Fig. 5.4(a) after removing the pixels above upper eyelids and below the lower eyelids (b) Pixels chosen by LM for the iris fitting (in white). The black pixels near the lower eyelids have been chosen, but then removed, since they are detected as part of the eyelids. (c) The fitted curves to the eyelids and the iris. The eyelids curves fit correctly in the area of great importance - near the iris*

dius to the expected range of the iris's radii. In our videos, the expected range of radii for the iris is between 96 to 144 pixels for over 95% of the patients.

After the iris has been detected, we use the Levenberg-Marquardt(LM) [41] algorithm by iteratively minimizing a non-linear function for fitting a circle. Before explaining how the LM minimization works, we note that it serves us for two distinctive tasks:

- On the first image, we use it to fine-tune the initial estimation of the iris.
- In the rest of the video, it serves as a form of iris tracking. For each image in the sequence, we would like to efficiently find the iris. Its location in the previous image serves as the initial guess for the LM algorithm. Since, we do not expect drastic movements of the iris between consecutive image frames, the LM algorithm converges quickly. The time-consuming RANSAC is used only for the first image, while for the rest of the sequence, only the much faster LM is being used.

The LM minimization works as follows: The initial estimation (x_0, y_0, r_0) is the one found for the iris in the previous image (or in the RANSAC step for the first image). We define the maximum error e to be a few pixels, usually around eight, and then bound our search area only to pixels that lie in the annulus formed by the circles of

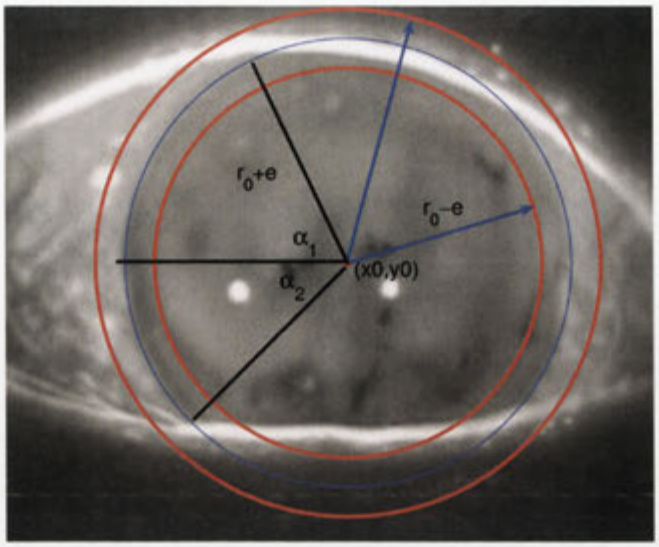


Figure 5.6: *Explaining the LM minimization: The image in the figure is from the sequence used in Fig. 5.4(a) and (d). The red circles show the bounded area that is searched for pixels in (5.2). The angles α_1 and α_2 are the angles of intersection between the eyelids and the iris. They limit the search area to the arcs between the intersection points. The strongest pixels found using (5.2) in this region (ROI_2), are used in the LM minimization step: (1) for the first frame to refine the current parameters of the iris. (2) to track the iris throughout the video. Therefore, the same ideas are used for the first frame and for the rest of the frames throughout the sequence.*

radii $r_0 - e$ and $r_0 + e$ and the center (x_0, y_0), as depicted in Fig. 5.6. However, since the iris is usually only partly visible because of the eyelids, we can further limit our search area. The angles α_1 and α_2 between the estimated iris and the upper and lower eyelids are computed. This creates a region bounded by two arcs for each side of the iris. We define the following cost function to estimate the strength of each pixel $(x_k, y_k) \in ROI_1$ in this region:

$$\psi(x_k, y_k) = M(x_k, y_k) + L(x_k, y_k) + |D(x_k, y_k)| \tag{5.2}$$

The term $M(x_k, y_k)$ is the magnitude of the pixel in the edge map. Using the magnitude only is not enough, as quite often noise on the conjunctiva has higher magnitude than the iris's pixels. To that end, we also compute $L(x_k, y_k)$, which is the length of the 8-connected segment the pixel belongs to. Our main assumption is that the iris consists of long connected segments at each half circle, while noise usually have shorter connected regions. The eyelids can also have long connected regions, but they are

usually not included in the search area because they were removed earlier. Finally, the term $|D(x_k, y_k)|$ is the absolute value of the Euclidean distance of the pixel from the iris. The value of $D(x_k, y_k)$ is negative when the pixel is inside the iris and positive when it is outside the iris and is defined as:

$$D(x_k, y_k) = \sqrt{(x_0 - x_k)^2 + (y_0 - y_k)^2} - r \tag{5.3}$$

Assuming that the iris has not moved significantly, the term will avoid digressing far from the current estimation. This is usually more important when searching for the iris throughout the sequence as large movements of the iris do not occur often. Bad fluorescein spreading or long connected edges parallel to the iris as seen in Fig. 5.5(a) in the right side, can be thought as part of the iris. In these cases, the term will penalize such fittings.

All the edge pixels are sorted according to the cost function $\psi(x, y)$ and the strongest 150 are taken for the LM minimization. The choice of this parameter is based on the fact that the diameter of the iris in most of the population ranges between $9mm$ and $13mm$. In our scaled image, it is correlated to a diameter of 96 pixels to 144 pixels. The visible area of the iris in our videos (not covered by the eyelids) is on average 85% of the iris area, so the number of pixels on the its perimeter range between 256 to 384. Therefore, on average we take every second pixel and denote this set of pixels as ROI_2.

The parameter to be minimized by the LM algorithm is the vector (x_0, y_0, r_0) (the iris coordinates and radius). At each iteration, the LM algorithm computes the sum of squares of the distance of each pixel in ROI_2 to the current estimate of the iris. In order to limit the penalty of outliers, we use the Huber distance [41], so the penalty above a threshold ϵ becomes linear with respect to ϵ. The penalty function is defined as follows:

$$f_d(x) = \begin{cases} x^2 & -\epsilon \leq x \leq \epsilon \\ \epsilon^2 + 2\epsilon(x - \epsilon) & x > \epsilon \\ \epsilon^2 + 2\epsilon(-x - \epsilon) & x < -\epsilon \end{cases} \tag{5.4}$$

At $x = \epsilon$, the function equals ϵ^2 and is continuous. The first derivative is also con-

tinuous at $x = \epsilon$ as $f'(x^-) = 2x = 2\epsilon$ and $f'(x^+) = 0 + 2x + 0 = 2\epsilon$. Similarly, the function is continuous and differential at $-\epsilon$. As the LM algorithm computes the sum of squares of the distances, the square root of (5.4) is used:

$$D_h(x) = \sqrt{f_d(x)} = \begin{cases} x & -\epsilon \leq x \leq \epsilon \\ \sqrt{(2x-\epsilon)\epsilon} & x > \epsilon \\ -\sqrt{(-2x-\epsilon)\epsilon} & x < -\epsilon \end{cases} \tag{5.5}$$

We notice that the function $D_h(x)$ is continuous and differential for all x. Finally, for each pixel $(x_k, y_k) \in ROI_2$, we would like to assign a penalty according to its distance $D(x_k, y_k)$ using (5.5). Thus, the LM algorithm minimizes the following function:

$$\min_{(x_0,y_0,r_0)} \sum_{(x_k,y_k)\in ROI_2} [D_h(D(x_k, y_k))]^2. \tag{5.6}$$

Using the Huber distance makes the penalty linear above the threshold ϵ, avoiding quadratic penalties to faraway outliers. The result after running the LM algorithm showing the chosen pixels is depicted in Fig. 5.5(b). The cost function of (5.2) keeps track of the iris even when a major movement happens and will converge to the correct location of the iris after a few video frames. We note that in such cases, the detection of the iris is incorrect for a few frames. However, our main goal is not iris detection, but analyzing the break pattern. Furthermore, we show in this chapter and the following one, that our segmentation technique handles correctly such cases by ignoring video frames that do not fit into the clinical definition of dryness.

Fig. 5.5(c) shows the final fitting results for the iris and the eyelids. If the detection of the iris fails, the thresholds used in (5.1) for the detection of the iris and the eyelids are adaptively updated. Failure in the fitting is declared when: (a) There are not enough pixels in the edge image used for fitting the eyelids. This can happen if the thresholds are too high; (b) The spreading of the pixels found by the LM algorithm over each half circle of the iris is not even; (c) The centers of the two opposing arcs of the iris are not closely opposite to each other.

5.3.1 Difference from other iris detection approaches

In chapter 3, several iris detection methods have been described. Although none of them has been used on fluorescein images, they still share some similarities with our method, such as circle detection and fitting a parabola to the eyelids.

One noticeable difference between other approaches and our algorithm is that we first detect the eyelids and only as a second step detect the iris. The reasoning behind is that the eyelids usually have the stronger edges in the image due to the fluorescein instillation, which makes the borders of the iris weak. Our approach is based on several unique key points:

1. Detecting the eyelids limits the search region for the iris in the first phase. We do not assume any prior information regarding the pupil or the iris being centered.

2. The intersection points between the eyelids and the estimated iris bound the search area in the LM optimization phase. Thus, we make no assumption regarding what percentage of the iris is not covered by the eyelids.

3. It is only important to detect correctly the eyelids in the iris area. Therefore, if the eyelids curve strongly in the left or right sides, it will usually have no effect on the region that borders the iris.

4. The LM algorithm is the only part of the algorithm that is used to detect the iris in the rest of the video (after the first image). Since the iris has to be detected on average in 100 video frames, the LM algorithm can be thought as a fast iris tracking algorithm. This idea is quite different from approaches that use the same method to detect the iris in consecutive images.

Finally, we show some of our segmentation results in Fig. 5.7. Six images were chosen from a data-set of 79 videos to depict the various cases. Each image in the figure is the first image of the relevant sequence (after the blink). Parts (a)-(c) of the figure show images where the iris and the eyelids are correctly detected. In part (d) the iris is not centered but is detected correctly. Part (e) shows a challenging image where the eyelids are irregular (folds grade 3), but the iris is segmented correctly. We

have managed to detect the iris in all but one image. This image is depicted in part (f), where only a small portion of the iris is visible in the right hand side. We note again that the iris fitting is only the first step of the dryness algorithm and correct segmentation results can be achieved even if the fitting is slightly wrong. In addition, the exact fitting of the eyelids is only important in the iris region (unlike in chapter 8 where we segment the tear meniscus).

5.4 Computing image Homography

After locating the iris in each of the images, it is possible to align the images over the iris area. The need for such alignment is demonstrated in Figures 5.1 & 5.10. In the previous section, we described how the iris is detected in each of the image frames and parameterized as a circle. Therefore, it is possible to perform a straightforward alignment from circle to circle. Such alignment has only three parameters: translation (t_x, t_y) and scaling (s). This is to due the fact that the circle has to be scaled equally in the x and y axes and is invariant to rotation (we will handle the rotation of the images later). The translation is due to movements of either the iris or the camera. The scaling occurs due to the clinician moving the camera forward or backward. Even when using a headrest, small movements of the chin result in change of scale. We align each image in the sequence to the first image. For example, assume the circle in the first image is parameterized as (x_0, y_0, r_0) and in the current frame as (x_1, y_1, r_1). The scaling is defined as $s = r_1/r_0$ and the translation vector as $[x_1 - x_0, y_1 - y_0]$.

The resulting aligned video is usually not aligned properly on the horizontal axis and when playing the video (of aligned images), it seems like it suffers from minor shaking. To that end, a second alignment is implemented.

The need for a second alignment is due to two main reasons. Firstly, there is still rotational ambiguity that needs to be resolved. The portable camera is free to move in the 3D space, so rotation is expected. Aligning the iris's circles (as in the first step) cannot resolve this ambiguity. Secondly, parts of the iris are not visible because of the eyelids. Thus, it is possible for a few different circles to fit to the iris.

The idea is to take the first aligned video as an input and then align by using

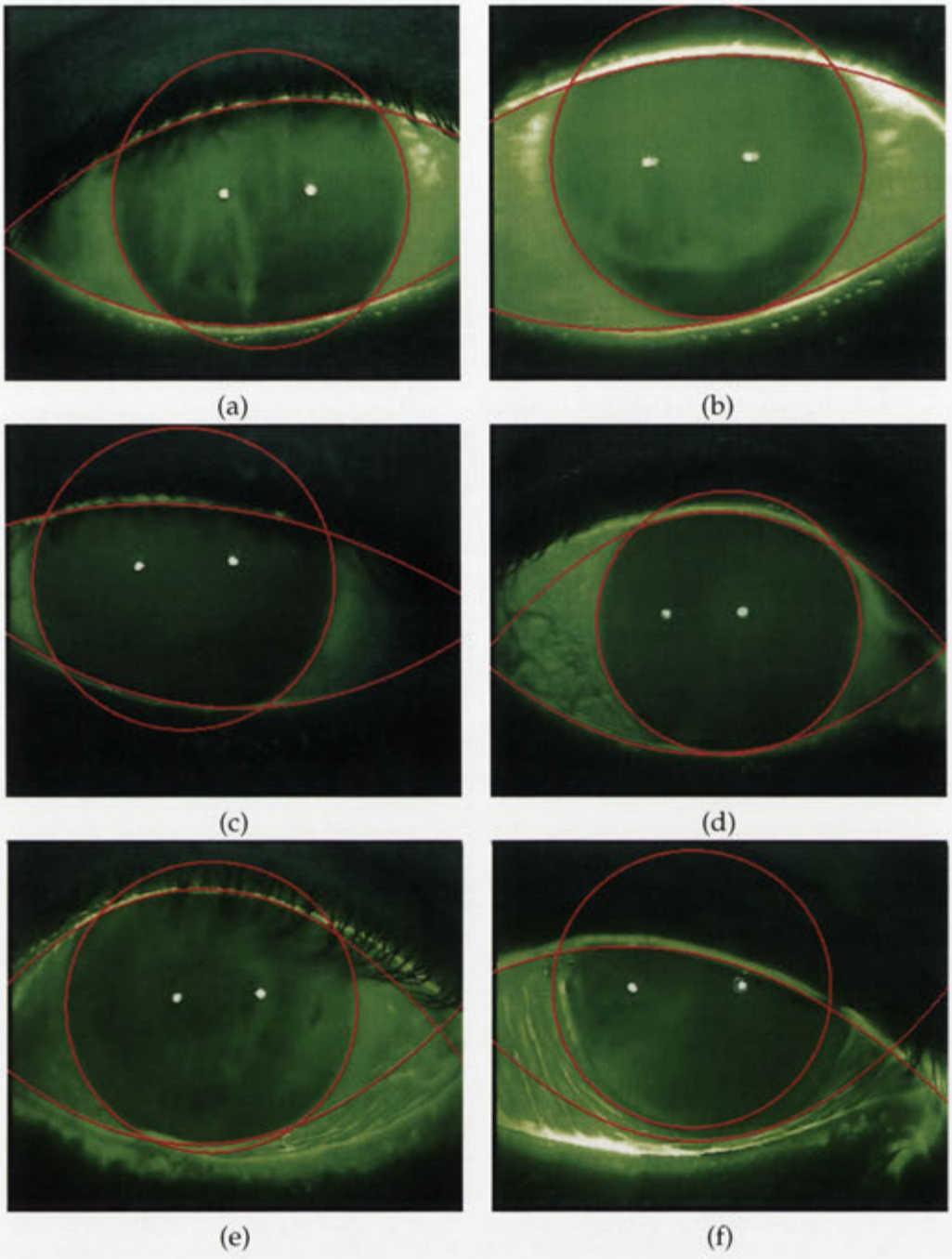


Figure 5.7: *Results of segmenting the iris. The figure shows 6 segmentation results of the iris. The iris is covered in varying degrees by the eyelids. In part (c), a major part of the iris is covered and a larger portion of the iris is visible in the right side compared to the left side. In part (d), the iris is not centered and has weak edges. In part (e), the eyelids are irregular and most of the iris is visible. Part (f) shows an image where the iris is incorrectly detected. The reason is the lack of edge information in the right hand side. It can be seen that the eyelids and regions on the conjunctiva have stronger edges than those of the iris. The detected iris follows edges inside the iris. Notice the difference in illumination between the figures, the patterns the fluorescein spreading creates over the iris (which result in spurious edges) and the evident noise created by the texture on the conjunctiva (especially in parts (d)and (e)).*

the grey levels of the iris area to find the best homography between images in the sequence. We would like to find the best homography H_x between two images which minimizes some criterion as detailed next. The result of applying this homography should be a sequence where the iris is located at the same location with the same radius throughout the whole sequence.

5.4.1 Our alignment procedure

We will now present our alignment algorithm which is based on the Lucas-Kanade algorithm (see section 3.2.1). Given the vector of images, every few consecutive images are treated as a block. The first image in the block is aligned to an image in the previous block and the other images are aligned to the first image in the block. The rationale behind alignment in blocks is to avoid accumulating error. If the images were aligned to the first frame in the sequence, large translation and scaling factors might accumulate, breaking the assumption of a small change from the previous image. The alignment in blocks produced better results than aligning to the previous image, as it is possible to recover from temporary errors. The alignment procedure is as follows:

1. Initialize the homography matrix $\mathbf{H}_0$ to be the identity matrix. Since we are interested in translation, scaling and rotation, the matrix has five unknowns. Define the initial region of interest (ROI) as the area that includes the iris and the eyelids. The eyelids differ in grey level from the iris and help the alignment process.

2. The homography $\mathbf{H}_0$ is used as our initial guess. A pyramid of scaled images for both images is built as seen in Fig. 5.8. The most coarse scale can be an image as small as 32×32 and the finest scale is the original image. For each scale, the best homography is found starting from the most coarse scale. Moving towards finer scales is done by rescaling the transformation parameters and the ROI.

   The usage of a scale-space approach has the benefit that we already have a rough estimate of the homography from the previous stage. In the most coarse scale, this estimate is usually quite accurate, so the algorithm converges quickly. When moving towards the finest scales, the change in parameters is small, so again the

computation is fairly simple and converges quickly. Furthermore, in the coarse levels, the noise has less effect as the images are highly smoothed. Finding the homography parameters for the finer scales converges quickly since the initial estimation $\mathbf{H}$ from the previous scale should be almost correct.

3. Compute the homography between images j and k by minimizing:

$$C(\mathbf{H}) = \sum_{x \in ROI} (I_x^{(j)} - I_{\mathbf{H}x}^{(k)})^2 \tag{5.7}$$

where x is a pixel in image j, $\mathbf{H}x$ is the transformed location of pixel x in image k, and $I_x^{(j)}$ and $I_{\mathbf{H}x}^{(k)}$ are the intensities of these pixels in the corresponding images.

One can see that (5.7) is exactly the same expression being minimized by the LK algorithm in (3.5).

4. Use the identity matrix if the homography computes a large scaling factor. This should only happen in the unlikely case when there is a drastic change between the images. By using the identity matrix, we basically assume that no change in the radius of the iris or coordinates has happened between the images and thus that the computed homography is incorrect. We explain below when this might happen.

5. The process is repeated for every image in the sequence.

In step 3, the best homography $C(\mathbf{H})$ is computed. This can be done by the LK algorithm. However, our minimization method uses instead the Levenberg-Marquardt (LM) non-linear optimization method. The LM algorithm is an iterative optimization method that alternates between Gauss-Newton and gradient descent methods. At each iteration of the LM algorithm the input vector is the set of parameters to be minimized (5 parameters) and the output vector (residuals vector) is some measurement of the accuracy of the current parameters. In our case, the output vector is of length of the number of pixels in the ROI and each entry equals to $(I_x^{(j)} - I_{\mathbf{H}x}^{(k)})$, which is just the difference in intensities between corresponding pixels in images j and k. The LM algorithm computes the sum of squares of the values in the residuals vector (5.7), which

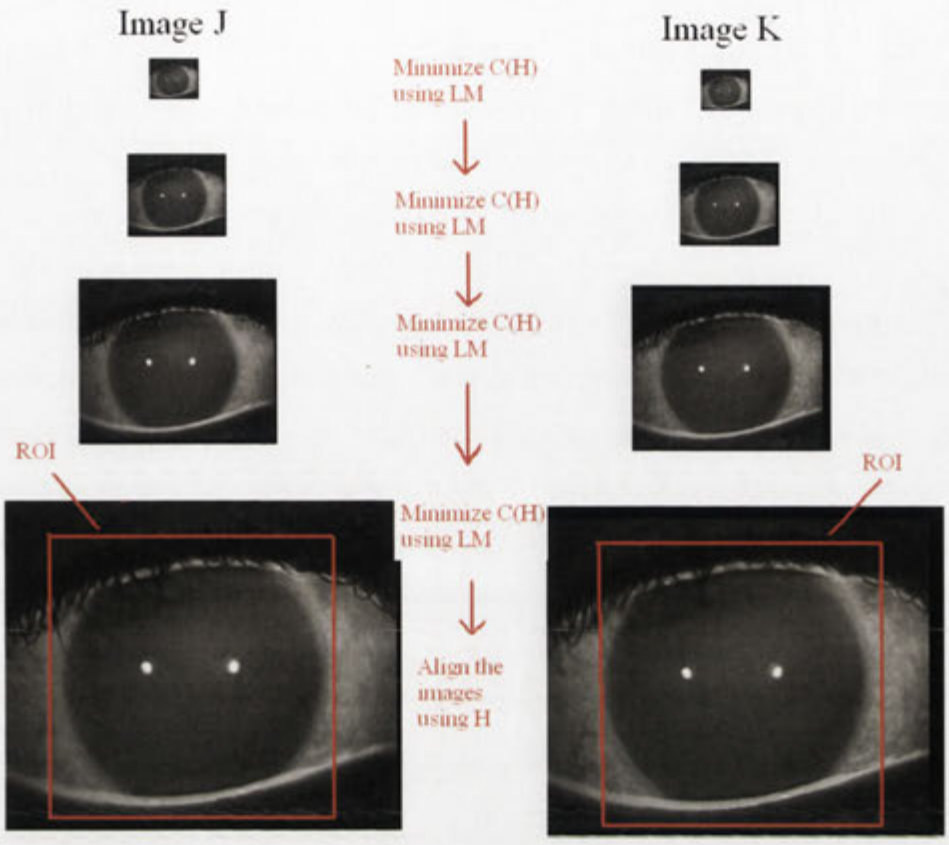


Figure 5.8: *Explaining the coarse to fine alignment using the LM algorithm. The images correspond to frame numbers 100 and 104 in the sequence used in Fig. 5.1. In this figure, 4 scales are used, where the most coarse image is 1/8 of the size of the original image. At each scale, LM is used to find the best homography H. Then the images and the homography are rescaled and the rescaled homography is used as the initial guess at the finer scale. The homography H found at the finest scale is used to align these two images and the process continues for the next frames.*

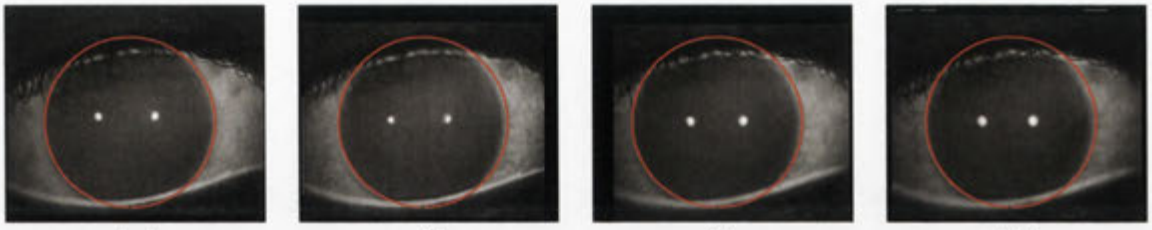
results in a value that defines the quality of the current homography. According to this value, the algorithm estimates an updated parameters vector (by modifying the damping term) to be minimized in the next iteration. The algorithm converges when the change in the parameters vector or the magnitude of the gradient of the term in (5.7) is below a threshold.

The term in (5.7) minimizes the difference in intensities between two images over the region of interest. Translation of the iris over the image and gradual scaling will not result in big differences between consecutive frames. However, when a movement

of the upper eyelids happens, such as further opening upwards or making an incomplete blink, a large area becomes immediately brighter or darker. The homography can incorrectly compute large or small scaling factors, where in practice no change in scale has happened. This is a known problem with the LK algorithm which struggles to converge to the correct answer when the change between two images is large. In such cases, the identity homography is used instead in step 4. If the opposite movement happens quickly then (5.7) will compute a correct homography again. In such cases, as in the sequence chosen for Fig. 5.10, where an incomplete blink is shown in Fig. 5.10(b)), temporarily using the identity matrix is correct and the alignment process will converge quickly to the correct iris parameters. If the eyelids move is a lasting one, such as moving upwards the upper eyelids without closing them until the next blink, the alignment error will be resolved once the alignment algorithm starts a new alignment block. The image from the new block is aligned to an image from a previous block (after the move has happened), basically ignoring the move and producing a correct alignment. Thus, the erroneous alignment is only for a few frames between two consecutive blocks.

To sum up, given two images, we first align the images using the location of the iris at each image. The mere purpose of this step is to perform a quick and easy alignment process, even if not perfect. This step provides initial guidance to the next alignment step and helps speeding up the minimization process. Then, we perform a second alignment based on a scale-space version of the LK algorithm and minimizing the cost function at each step with the LM algorithm. The alignment starts from the coarsest scale towards the finest scale at both images. Upon converging, the homography parameters and the region of interest are rescaled. The transformation is applied on the image to be aligned in the finer scale. Then we run the LM algorithm again at successively finer scales until reaching the finest scale. The output parameters of the finest scale define the best homography. The process repeats for the next two images of the sequence.

The importance of the alignment step should not be underestimated, but it is difficult to show the results of an alignment unless it is shown in a video. Fig. 5.9 demonstrates the effect of the alignment on the sequence of Fig. 5.1. The iris was automati-



(a) (b) (c) (d)

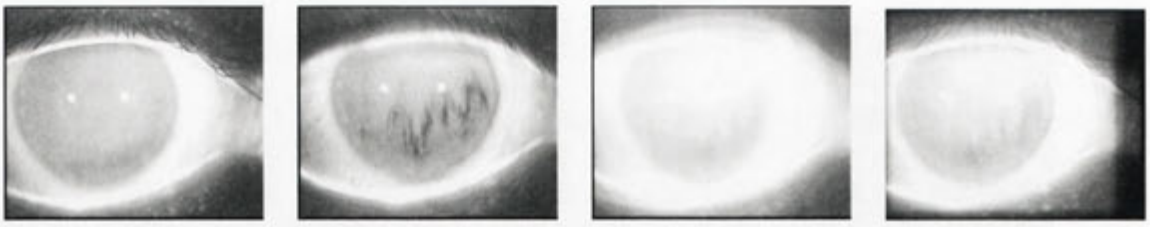
Figure 5.9: *Result of the aligned video of Fig. 5.1. In each of the images, the iris is drawn at the same location as it is in the first image. Compare the result here with the one in Fig. 5.1 to see the effect of the alignment.*

cally detected in the first image in the sequence, see Fig. 5.9(a). Then, we took three additional images from the sequence (figs (b)-(d)), and manually plotted a red circle at the same location as it was plotted in the first image. We can see that the alignment process worked correctly since the red circle surrounds correctly the iris in each of the images (with a very small misalignment in the upper right hand side). Compare these results with those of Fig. 5.1 to see the importance of the alignment.

Fig. 5.10 shows the averaged images over a sequence which has a break. The image in part (a) is the first image in the sequence and in part (b) the last one. The non-aligned image in part (c) is completely blurred and cannot provide information regarding the dry areas. It is expected since the camera is hand held and the patient's gaze can move. The averaged image of part (d) has much less blur in the iris area as a result of a successful alignment.

5.5 Segmentation of the dry areas

After the video has been aligned, it is possible to detect the dryness symptoms. The general idea is to scan the vector of aligned images and gather local information regarding the change of intensity of each pixel in the iris and global information to relate each pixel to the iris properties. An outline of the segmentation algorithm is given below and elaborated in the next subsections. It can be roughly divided into two main parts: 1) creation of the dryness image $\tilde{I}(x, y)$, which is the base for further analysis, at lines 6-9. 2) temporal analysis of the video at lines 10-18: a new measurement called



(a) (b) (c) (d)

Figure 5.10: *Since the camera is hand held and the iris can move, it is necessary to align the video: (a) image after a blink (b) image before the next blink. The sequence is chosen to show the effect of an incomplete blink, as a series of 3 brief incomplete blinks happened before the full blink. The averaged image of 96 frames from the (c) non-aligned video (d) aligned video. The non-aligned result is blurred, and the iris spans over a large area. It also shows that the dry areas cannot be detected without aligning the images or by just taking the difference between images (a) and (b). The aligned result still shows some of the dry areas that formed on the lower part, and the upper eyelids are intact even though an incomplete blink has occurred.*

DEBUT, which is correlated to the clinician's BUT, is defined and computed (see section 5.5.2) and each break area is analyzed for its temporal progress. The algorithm is given as follows:

Algorithm *Find Dry Areas*

Input: A vector of n aligned images between two consecutive blinks.

Output: Dryness images of corneal break and black line. Digital Electronic Break up Time (DEBUT).

1. Initialize DEBUT: $\forall_{(x,y)} T(x,y) = n+1$ (No break for all pixels).
2. Given the first image I_1, define ROI_3 as the set of all the pixels of the iris.
3. Divide the set ROI_3 into 3 subsets: upper eyelids, lower eyelids and corneal area.
4. Calculate the average intensity for each of the three areas. ($*$ To be used later for the creation of the dryness images and the computation of the black line. $*$)
5. Repeat lines No. 2-4 for the last image I_n.
6. **for** each pixel $(x,y) \in ROI_3$
7. Calculate the terms used to estimate the pixel's degree of thinning: $F(x,y)$ and $B(x,y)$ ($*$ See below $*$).
8. Calculate the final dryness value $\bar{I}(x,y)$ ($*$ See below $*$).
9. Find black line in the first and last image.
10. **for** each pixel $(x,y) \in ROI_3$
11. **for** each image $t \leftarrow 1$ **to** n

12. Compute the pixels' current dryness intensity $\bar{I}_t(x, y)$ based on the t first images.
13. **if** $\bar{I}_t(x, y)$ is higher than the break threshold B_T **and** $T(x, y) > t$
14. **then** $T(x, y) \leftarrow t$ (∗ Update the DEBUT for the pixel ∗)
15. **if** a pixel with a break $T(x, y) < t$ further decreases in intensity
16. **then** $T(x, y) \leftarrow t$
17. Analyze dryness image for individual break areas and their progress.
18. Find DEBUT and cause (corneal dryness, upper or lower eyelids) and statistics.

5.5.1 Producing the dryness images

The dryness image is an image which shows the degree of dryness for each pixel in the iris area. An intensity value $\bar{I}(x, y)$ is calculated for each pixel, which is related to the degree of thinning . The higher the value of $\bar{I}(x, y)$, the drier it has become. A pixel with an intensity above B_T is a pixel with a full thickness break (a break). The value for B_T has been chosen to be close to the maximum intensity of a pixel for clarity of presentation. Therefore, in the dryness images, see example in Fig. 5.11(e), break areas appear as very bright regions. Areas which depict some thinning of the tear film are grey and areas with no thinning are black.

Once the value for B_T has been fixed, the rest of the computations are adjusted according to it. Therefore, on one hand, the value of B_T has a significant importance as it determines which pixels have a break. On the other hand, the actual value is not that important as it was set early on, and all computations, such in (5.8) and (5.9), were adjusted to its value. For example, the constant T_D in (5.9) has been set using learning videos. However, the learning process is dependant on the pre-defined value of B_T, as in order to perform the learning correctly, it needs to know when a break has happened.

In line 4, the average iris intensity in the first image, A_{I1} is computed. The brightest $1/8$ pixels are omitted from the calculation, as errors in the alignment, the bright two circles in the iris area (see Fig. 5.10 for an example) and the inclusion of parts from the eyelids (very bright pixels) can bias the result. The choice of this value ($1/8$) is based on approximating the ratio between the wanted and unwanted pixels in images taken

immediately after a blink. The dryness image is computed as follows:

$$\tilde{I}(x,y) = F(x,y) + B(x,y) \tag{5.8}$$

1. The term $F(x,y)$ is the mean difference in intensity between the first n_d images and the last n_d images multiplied by a normalization factor:

$$F(x,y) = \frac{(T_D/A_{I1})}{n_d}\Big[\sum_{k=1}^{n_d} I_k(x,y) - \sum_{k=n-n_d+1}^{n} I_k(x,y)\Big] \tag{5.9}$$

   This term is the more important one out of the two in (5.8). The reasoning for using differences in intensities is straightforward, as dry areas have to appear darker in images towards the end of the sequence compared to images in the beginning of the sequence. We use a value of $n_d = 4$, so $F(x,y)$ is not based on a single image and thus less sensitive to a temporary misalignment of the pixel.

   As the average intensity for the iris can be as high as 150 or as low as 60, $F(x,y)$ is computed relative to the iris average intensity by dividing by A_{I1}. A low value of A_{I1} results in a fast increase of $\tilde{I}(x,y)$, for every small decrease in the pixel's intensity. A high value of A_{I1} results in a slower increase of $\tilde{I}(x,y)$ for changes in the pixel's intensity. Therefore, images which are initially darker are more sensitive to small changes in intensity throughout the video.

   The second constant T_D normalizes the degree of thinning relative to the constant B_T. Its value has been set through learning by comparing to manual segmentations of dry areas and the BUT values provided by an optometrist. Its value is set to a value higher than the brightest iris average: $(T_D/A_{I1}) > 1$.

2. The bonus map $B(x,y)$ relates the pixel initial intensity to the iris initial average intensity. It is used to compensate for pixels whose initial value is lower or higher than the average value. In order to explain this idea, let us simplify matters and assume three pixels on the iris with the following decrease in intensity: $p1 : 120 \rightarrow 90$, $p2 : 100 \rightarrow 70$ and $p3 : 140 \rightarrow 110$. All three pixels have the same absolute change in intensity, therefore $F(x,y)$ is the same for all three pixels. However, a decrease in intensity from 100 to 70 is perceived by the

clinician as more meaningful than a decrease from 140 to 110. This might be attributed to the fact that in low contrast images, break areas seem dark enough even when there is only a small decrease in intensity. Assuming $A_{I1} = 120$, pixel $p1$ has a bonus of 0, pixel $p2$ a bonus of 20 and pixel $p3$ a bonus of -20. In badly illuminated images, this term can be quite meaningful, but generally less than $F(x, y)$.

We note that setting the parameters of (5.8) in a way that will match the clinician at all times is almost impossible. This is due to the disparity between the BUT values given by different clinicians and sometimes even by the same clinician at different times (see Section 7.1.1). To that end, we introduce the DEBUT value in the next section.

5.5.2 DEBUT computation

As described in the introduction and also shown in chapter 7, the BUT value varies between clinicians. Therefore, we have found it appropriate to define a new value that is correlated to the BUT measured by the clinicians, but is defined in terms of image processing and the temporal progress of the break. This value is called the Digital Electronic Break Up Time (DEBUT), and is closely related to the way the dryness image is built. This value is computed such that it does not depend on the clinician and is robust to changes in illumination and movements of the iris. It is defined and explained as follows, and also in Fig. 5.11:

1. A pixel (x, y) in the image is considered as a full thickness break at time t when its intensity in the dryness image at time t ($\bar{I}_t(x, y)$) is over the pre-computed threshold, B_T. The intensity value is calculated in line 12 in the same way as for the final dryness image in (5.8), however, using only images up to the current frame t. For example, $F_t(x, y)$, computes the difference using the first n_d images and the images between $t - n_d + 1$ to t.

   If the pixel's computed dryness intensity is above the threshold B_T, we update the DEBUT image T (see line 1) and assign the current time to this pixel ($T(x, y) = t$) at line 14. This records the first time the pixel is above the break

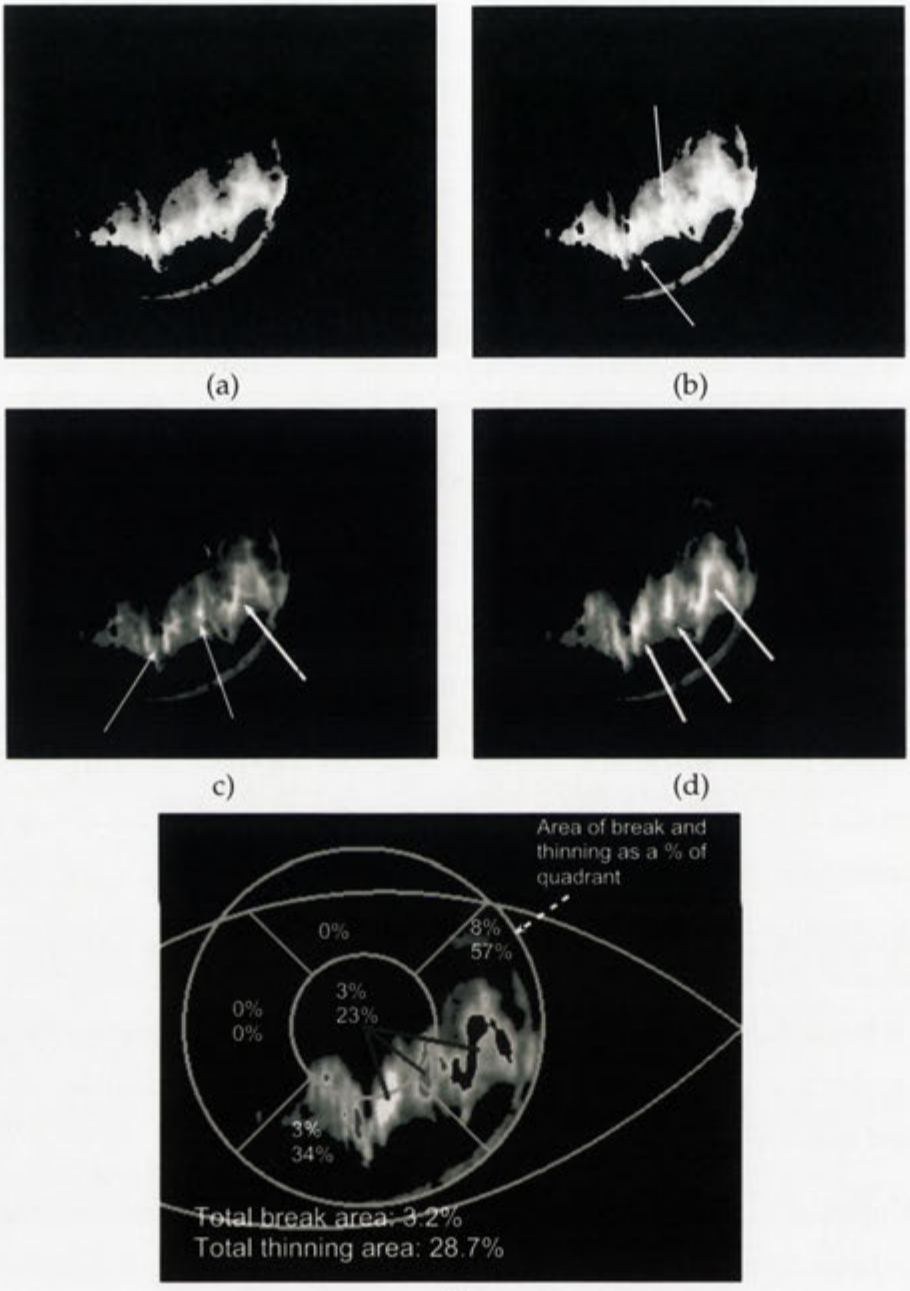


Figure 5.11: *DEBUT explained. The figure shows how the dryness image is built and the DEBUT is computed. The sequence is the same one that was used in Fig. 5.10 and consists of 224 frames (9 seconds). The computed DEBUT is 135 image frames, 5.4s. The analysis of the individual break regions is given in Table 5.1. (a) Dryness image after 75 frames. There are no pixels of full thickness break yet. (b) The dryness image after 105 frames. The arrows point to areas of full thickness break. However, (5.10) still does not hold. (c) The dryness image after 135 frames. The thick arrow points to the area r_j that complies with (5.10). (d) The dryness image after 205 frames: (5.10) holds now for a few individual areas. Other areas have progressed as well to form in total 4 individual break areas. (e) The final dryness image. The lines point to the center of each individual break area.*

threshold (thus having a break). Clinically, once an area is discovered as a break, it should stay as a break until the following blink. Therefore, $T(x, y)$ should not be updated more than once in perfect conditions. We discuss this issue later.

The arrows in Fig. 5.11(b) point to pixels of full thickness break at time $t = 105$. The first frame to have one or more pixels above B_T is the 75th frame (Fig. 5.11(a)).

2. When the pixel's computed dryness intensity is over the threshold B_T, it is not expected to have further significant decrease in intensity. This is since areas devoid of fluorescein cannot have further breaks (or they are not expected to become any darker). If a pixel (x, y) shows significant decrease in intensity at time $t > T(x, y)$, it is adjusted to have a later BUT. If a pixel eventually does not appear in the final dryness image, we assign $T(x, y) \leftarrow n + 1$.

3. An **individual area** r_j^t is defined as the set of all pixels of the j-th 8-connected neighboring region at time t in the dryness image $\tilde{I}_t(x, y)$, such that all pixels in the region are assigned as full thickness break. Finding the connected areas can be done easily by analyzing the DEBUT image T and searching for connected regions that all pixels (x, y) in the region have a value of $T(x, y) <= t$. In Fig. 5.11(c), three major individual areas are detected.

4. The break up time is defined as the minimum time t passed since the blink such there exists an individual area r_j^t of size larger than the threshold A_{DEBUT} (explained below). In another words, we are looking for a connected region of a certain size at time t, that all pixels in the region have a full thickness break (a break). The DEBUT is defined as follows:

$$DEBUT = \min_t \left\{ \exists r_j \left[\left(\sum_{(x,y) \in r_j^t} \delta_T(x, y) \right) > A_{DEBUT} \right] \right\}. \tag{5.10}$$

The indicator function δ_T is defined as:

$$\delta_T = \begin{cases} 1 & T(x, y) \leq t \\ 0 & \text{otherwise} \end{cases} \tag{5.11}$$

The term $\sum_{(x,y)\in r_j^t} \delta_T(x,y)$ sums the number of pixels in a connected region that have a break at time t. Therefore, the minimum t is related to the first time since the blink, that a region has enough break pixels. Fig. 5.11(c) depicts the dryness image, when this requirement holds for the first time. The current frame number, which is 135 in that case, is assigned as the DEBUT value. In Fig. 5.11(d) the break areas have progressed to have a total of 3 individual areas.

$\mathbf{S_{iris}}$ is defined as the total number of visible pixels in the iris (not hidden by the eyelids) at the first image of the sequence. After the alignment, the size of the iris should be the same at all frames. It is possible that parts of the iris will be occluded only at some frames, but it will not change $\mathbf{S_{iris}}$ significantly. Finally, we set the portion of iris pixels $\mathbf{T_S}$ to 0.0025. Then the minimum size for a break A_{DEBUT} ranges from:

$$90 \leq A_{DEBUT} = S_{iris} T_S \leq 140 \tag{5.12}$$

This value is chosen to make sure that the DEBUT is computed correctly even when there are some misaligned pixels. It can be correlated to the actual area in the iris: as mentioned in section 5.3, the diameter of the iris in over 95% of the population is between $9mm$ and $13mm$ and the visible area of the iris in our videos is on average 85% of the iris area. Therefore, the minimum area that is required for a break ranges between $0.0162mm^2$ to $0.0338mm^2$. This value makes sense, as smaller areas can hardly be detected by the clinician. However, as we found out through experimental tests (see chapter 7), it is sometimes necessary to detect even tiny breaks. A small value of A_{DEBUT} will make the algorithm very sensitive to noise and in some cases, it will incorrectly detect an early DEBUT. In chapter 6, an improved algorithm for computing the dryness image is presented, which is less sensitive to outliers. See also the discussion in section 6.6 about reducing the A_{DEBUT} value.

In the dryness images of Fig. 5.11(e) and the third row of Fig. 5.12, the break areas are highlighted on top of the dryness image. We divide the corneal area into 5 areas (see Fig. 5.11(e)) in a similar way to the CCLRU standards (Contact Lens Research Unit) in [105]. The area of thinning and break in each quadrant (Fig. 5.11(e)) is calculated and reported back to the clinician to be used in a follow-up inspection. The

Location	Shape	Size	Progress
Inferior	Vertical	Medium	Medium
Nasal	Vertical	Small	Slow
Nasal	Pool	Small	Slow

Table 5.1: *Analysis of the break regions of Fig. 5.11(e) from top to bottom according to the region's center (The arrows in Fig. 5.11(e) point to each region's center).*

fourth row in Fig. 5.12 shows graphs of the progress of the dry area as a percentage of the iris area. The blue graph depicts the area of the thinning and the pink graph depicts the area of the break as function of the time passed since the blink. The thinning graph provides general information about the thinning of the tear film. The graph in Fig. 5.12(a) shows a steady fast progress throughout time. The break graph lies on the x-axis until approximately 6.3s, which means that there are no pixels of full thickness break until that time. In other words, (5.8) is less than B_T for all pixels. Once the graph is lifted above 0 percent, it quickly satisfies (5.10) as the area of break grows steadily as seen in the dryness image. The graph and the DEBUT value are highly correlated. However, the break area depicted in the graph at the time of the DEBUT can be higher than (5.12), since breaks can progress simultaneously in different regions. These graphs provide global information about the whole iris area in regard to the thinning of the tear film and are useful visual tools for the clinician.

5.5.3 Analyzing individual breaks

After the dryness image is created, it is of interest to analyze the dry areas to add qualitative information regarding the break. The analysis is divided into two main key factors:

1. Analyzing each individual connected region, r_j for its size, location, shape and orientation.

2. Analyzing the progress of the size of the break - for each individual region, how fast the break area increases after the break has happened.

Analyzing each individual area is performed over the final dryness image. For example, the analysis of the dryness image in Fig. 5.11(e) is given in Table 5.1. Each

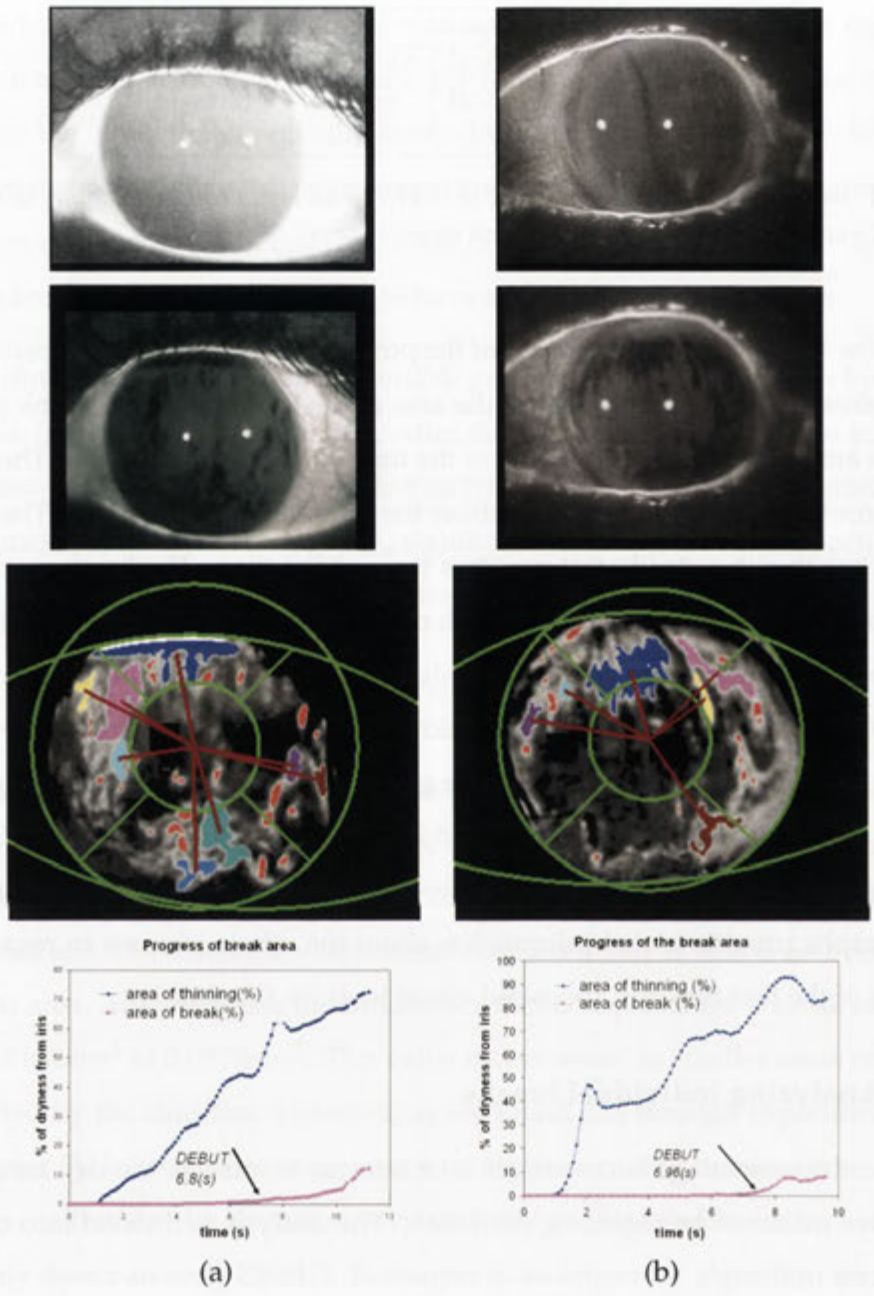


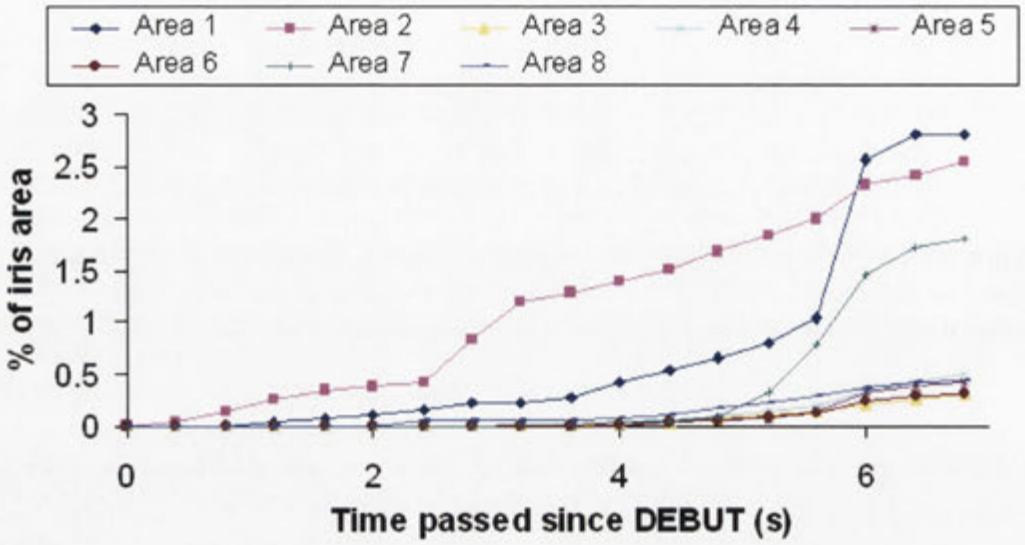
Figure 5.12: *Dryness detection results. Images from top to bottom: the image immediately after a blink; The image before the next blink (from the aligned video); the dryness image (cropped). The colored areas show individual regions of full thickness break. The other areas show the different degrees of thinning (the brighter, the drier it has become).; Graph of the evolution of the dryness: the graphs in pink and blue show the area (as a percentage of the iris's area) of break and tear film thinning respectively. The graphs are presented as a function of the time passed since the blink (x-axis). The decrease in the area of thinning (after approx. 7s in the left graph and 2s in the right graph) happens due to pixels that are misaligned temporarily and bias the sum of (5.8). The computed DEBUT for the left column is: 6.8s, while the range of BUT values given by four clinicians is 2.6s-6.9s.*

Location	Shape	Size	Progress
Center	Horizontal	Small	Slow
Nasal	Vertical	Medium	Medium
Temporal	Slanted	Big	Medium-Stabilize
Superior	Circular		Fast
Inferior	Pool		Fast-Stabilize

Table 5.2: *The list of possible outcomes when analyzing the dryness image in terms of individual break regions and progress of each region. Stabilize means that the progress of the break is Medium or Fast but then the progress slows down towards the end of the sequence.*

8-connected area is examined for its shape from the following list: horizontal line, vertical line, slanted line, circular and pool, which is used to describe any other shape [10]. Detecting the shape is done by first applying a morphological close on the image with a disc of radius 2 pixels. This should connect areas that otherwise might not be connected, usually due to intensity miscalculation of single pixels. Then we fit an ellipse to the region and use the ratio between the major and minor axes to determine the shape. We use the orientation of the major axis to decide the orientation. Since it is possible to have many individual areas, we take only those that comply with (5.12) in terms of the minimum break size. However, circular breaks are usually smaller and very local as they tend not to develop throughout the sequence. Thus, if the break area is detected as a circular, it will be taken even if it is smaller than the minimum size in (5.12). To simplify matters for the clinician, each individual area is plotted in a different color and a synopsis is provided using terms such as "big" or "horizontal". Table 5.2 summarizes the possible outputs for the clinician and the table at Fig. 5.13 shows the report for the dryness image of Fig. 5.12(a).

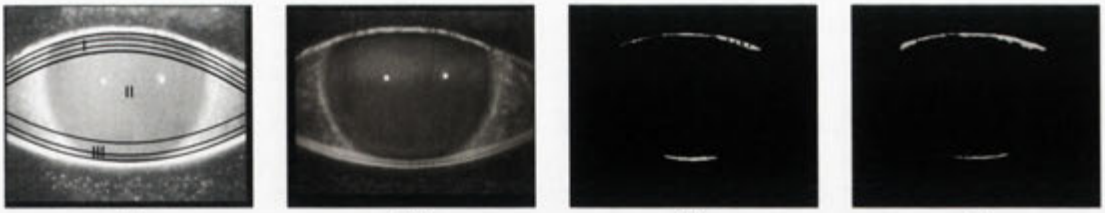
Analyzing the progress of each individual break requires scanning the dryness images at times $t \geq DEBUT_j$, where $DEBUT_j$ is the break up time of area j. The idea is to compute the area of the break at these times, and to evaluate the change in size in terms of speed of progress. The starting point is the analyzed area r_j in the final dryness image. Then, scanning backwards in time, we take the dryness image every few frames and find the size of the corresponding region. It is possible for a region to split into a few separate regions during the scan, so the joint area's size is used. For each individual region r_j, a vector $\mathbf{V_{jt}} = (a_{j;t_0}, \ldots, a_{j;DEBUT_j}, \ldots, a_{j;n})$ of the



No.	Location	Shape	Size	Progress
1	*Superior*	*Horizontal*	Medium	Fast
2	Temporal	Vertical	Medium	**Medium**
3	Temporal	Pool	Small	Slow
4	Temporal	Pool	Small	Slow
5	Nasal	Vertical	Small	Slow
6	Nasal	Pool	Small	Slow
7	Inferior	Pool	Medium	**Fast**
8	Inferior	Pool	Small	Slow

Figure 5.13: *Graph of the progress of regions of full thickness break and their corresponding analysis. The graph is for the sequence shown in Fig. 5.12(a). There are in total 8 areas of break as depicted in the dryness image in the third row of Fig. 5.12(a) and they are numbered from top to bottom. The colors of the graphs in the figure correspond to break regions with the same color in the dryness image. Three major break areas develop into large areas and the rest start at a later time and stay small. It is of interest that area 7 starts to form only 3 seconds after area 2 but has faster progress. Also notice that area 1 has a horizontal shape and is formed in the superior region. This could be related to a reservoir related break as described in section 5.5.4.*

region's size is built, where n is the number of frames in the video, $a_{j;t}$ is the size of the region at time t and t_0 is the first time the region has more than zero break pixels (notice that $t_0 \leq DEBUT_j$). Using linear regression, a least-squares fitting $\mathbf{f_j}(\mathbf{t}) = a_0 + a_1(t - t_0) + a_2(t - t_0)^2$ is found to $\mathbf{V_{jt}}$, where t is the time passed since the blink. The change in the region's size at time t is computed by applying the first derivative: $\mathbf{f_j'}(\mathbf{t})$. The change is divided by the minimum number of pixels for a full thickness



(a) (b) (c) (d)

Figure 5.14: *Results of detecting reservoir related dryness: (a) The image immediately after a blink. The three areas described in the algorithm in Line 3 are numbered. The curves show the regions searched for the black line as explained in section 5.5.4. The curves for the upper eyelids from top to bottom define the: eyelids curve, search region for the black line in I_1 (first image after the blink), search region for the black line in I_n (image before the next blink), the area used to compute the region's average m_{u1}. (b) The image before the next blink. (c) The black line after the blink. (d) The black line before the next blink. The brighter the area the stronger is the thinning of the tear film. In this video, only areas in the immediate proximity to the eyelids become dry and the break is a result of thinning near the upper eyelids only (no corneal break). The increase of the area of the dryness near the upper eyelids (see the difference between part (c) to (d)) is of clinical importance.*

break using (5.12):

$$\mathbf{f_j'(t)}/A_{DEBUT}. \tag{5.13}$$

This ratio is an indicator of the speed of the progress at time t and is used to define the progress as Slow, Medium or Fast (see Table 5.2). A stabilizing region is defined as a region that: (1) the slope is monotonically decreasing towards the end of the sequence, such that exists $k > 2$ satisfying $f_j'(n-k) \leq f_j'(n-k+1) \leq \ldots \leq f_j'(n)$. (2) the change is lower than the minimum break size: $f_j'(n-k') < A_{DEBUT}$ for all $k' = 0 \ldots k$. If the change is bigger than A_{DEBUT}, then the region is considered as growing, since the increase in the region's size is still significant. These definitions can be adjusted to different standards of measuring the progress. Fig. 5.13 shows the progress of each individual break as function of the total iris area. It is noticeable how some regions grow much faster than others.

5.5.4 Black line detection

The clinical importance of detecting the black line has been discussed in section 2.1.1. If the black line exists, it will always appear very close to the eyelids and should be noticeably darker than the immediate surroundings. Its detection combined with the

dryness images can provide valuable information to the clinician regarding the cause of the dryness (reservoir related, for example) and what treatment is needed. The reasoning behind performing a separate detection for the black line is:

- It clinically requires a different treatment.
- Accurate detection of the black line is difficult due to the its thin size and the proximity to the eyelids. Every misalignment near the eyelids results in an immediate miscalculation of the pixel's dryness using (5.8). Since the eyelids are usually brighter than the iris, the term $F(x, y)$ can produce a value of extreme dryness, even if no dryness exists (as the difference between the bright pixels to the dark ones can be high).
- The black line cannot always be discovered by looking at temporal changes of dryness because it might not be visible at all times. For example, the black line can be visible at the beginning of the sequence, but not towards the end of it if the patient closes a bit the upper eyelids.

The ground for the detection of the black line is prepared in lines 3-5, where the averages for the regions near the eyelids are calculated for I_1 and I_n (the first and last image of the sequence). They are denoted by m_{u1} and m_{l1}, for the upper and lower eyelids respectively (for the first image). These regions are depicted in Fig. 5.14(a), numbered by I and III and are bounded by the eyelids curve and the furthest curve from the eyelids. These regions contain a significant part of the iris, such that the difference in intensities of the black line from the iris is noticeable (if it exists). In most cases, we expect to have quite a thin break line (if at all) after the blink and possibly a bigger area before the next blink. To that end, the search area for the black line after the blink is bounded by the eyelids and the next curve after the eyelids. The search area before the next blink is doubled and is bounded by the the eyelids and the second curve from the eyelids. To conclude, the goal of these regions is to be able to detect a darker region relative to its immediate surroundings and not necessarily relative to the whole iris.

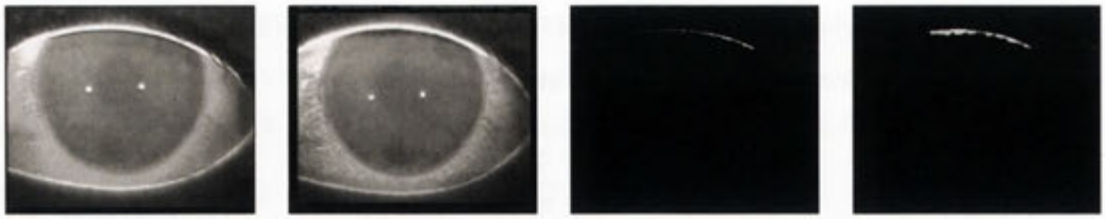
A pixel $I_1(x, y)$ near the upper eyelids is segmented if there is noticeable difference in intensity compared to the region's average intensity:

$$\lambda_b m_{u1} > \frac{1}{n_d} \sum_{k=1}^{n_d} I_k(x, y). \tag{5.14}$$

Therefore, the average pixel intensity in the first n_d images (see also definition of $F(x, y)$ in section 5.5.1) has to be lower than the region's average (m_{u1}). The value of λ_b is computed using the histogram of intensities in this region. Since the averages m_{u1} and m_{l1} are computed for a small region, the value of λ_b is usually close to 1 (but less than 1). Accurate detection of the eyelids (see Sec. 5.3) over the iris area is crucial in order not to include the bright pixels of the eyelids. The equation handles changes in intensity between images and between the upper and lower eyelids. It is also sensitive enough to discover thin break lines (see Fig. 5.14). Similar ideas are used for the lower eyelids and for both eyelids in I_n.

If a pixel is detected as part of the black line, we associate a dryness value to that pixel. This value is based on the same concepts discussed for computing the dryness image. Even if we have detected a region as part of the black line, its clinical importance depends on the dryness value of its pixels. Therefore, we assume that a pixel in the black line is break related (and thus clinically important) if its value is higher than B_T. Again, this idea is very similar to the one we have used for computing the DEBUT. The minimum size of a region to be considered a reservoir related break is the same as in (5.12), however it is not required to be contiguous. The removal of the continuity requirement is due to its small size and the likelihood of small holes in the segmentation due to its proximity to the eyelids.

We produce two dryness images for I_1 and I_n, corresponding to the black lines at times $t = 1$ and $t = n$ respectively. The brighter the pixel, the dryer it has become. By alternating between the images, the clinician can see the changes in the area of the black line and direction of development. For example, by alternating between images (c) and (d) of Fig. 5.14, the clinician can see that the black line in the lower reservoir existed after the blink and also before the next blink, while the upper black line increased in size.



(a) (b) (c) (d)

Figure 5.15: *Another result of detecting reservoir related dryness: (a) The image immediately after a blink. (b) The image before the next blink (from the aligned video). (c) The black line after the blink. (d) The black line before the next blink. The brighter the area the dryer it became. The visible increase of the area of the dryness near the upper eyelids is of clinical importance. There is no evident thinning near the lower eyelids.*

Finally, we distinguish between corneal break and reservoir related break (black line). Corneal break is defined as a sequence that has a DEBUT value ($DEBUT < n$). Reservoir related break in the lower or upper reservoirs is defined when a black line exists with a break. If a black line is detected, we report a break near the upper or lower tear reservoirs (or both), which probably caused the imminent blink. If the black line exists in conjunction with a corneal break ($DEBUT < n$), it is reported as a secondary cause. In the sequence of Fig. 5.14, black lines with a break exist for both the upper and lower eyelids and are reported as the main causes. The sequence of Fig. 5.15 has a thin black line near the upper eyelids after the blink and a thicker black line with a break near the upper eyelids before the next blink.

5.6 Conclusions and discussion

In this chapter we have presented our main algorithm for dryness detection and segmentation. We provide a quantitative evaluation of the degree of the thinning and break of the tear film. This includes the area, location, shape and progress of the break. Reservoir related breaks (black lines) are detected separately and are reported back to the clinician and can be linked to different dryness origins. Our graphical analysis provides useful information to the clinician regarding the origin of the dryness and can be recorded for a follow-up. It therefore surpasses the current clinical analysis, which is mainly based on the subjectively measured BUT.

We have defined the DEBUT as a new way to automatically measure the BUT. This new measure is computed quickly and is reported back as part of our analysis. Experiments to evaluate the robustness of this value are discussed in chapter 7. We have explicitly named this value differently than the BUT as we do not aim to match the clinician's value at all times. The main drives behind its definition are: (1) to use image processing techniques to be as close as possible to the BUT values provided by the clinician; (2) to compensate for the large inter-observer variance of the BUT measured by clinicians (demonstrated in chapter 7); (3) to define a value which is robust to uneven illumination, the degree of thinning of the tear film, the break size and the speed the break develops.

An extension of the algorithm to include spatial and temporal information is presented in the next chapter. This extension treats the whole video as one 3D volume and incorporates a monotonic constraint on the dryness evolution. This extension overcomes one of the major hardships of the algorithm: handling outliers in the dryness segmentation process and the DEBUT evaluation. Alignment errors of even one pixel near the eyelids can affect the segmentation result. We show that the extended version handles outliers better.

Some issues still remain open towards a complete dry eye system. The detected DEBUT can be biased by uneven fluorescein spreading or patients that gradually open their eyes throughout the test. In the first case, a break can be detected too early, as the shifting of the fluorescein creates artificial dry areas. In the second case, the break might be detected late because the new revealed areas in the eyelids are not immediately searched. Both cases can be handled by designing dedicated checks: The eyelids movement could be detected by measuring the visible region of the iris after the alignment throughout the sequence. The fluorescein spreading can be tracked by looking for specific patterns that move upwards immediately after the blink. Another way is to better control the amount of fluorescein that is instilled is using DET strips.

Chapter 6

Enforcing monotonic temporal evolution

In this chapter, we present an extension to the segmentation method described in the previous chapter. The idea is to use the relationship between pixels to provide a more robust segmentation which is less sensitive to noise and errors in the alignment process. The segmentation algorithm from the previous chapter segments each pixel individually, regardless of its location in the iris, its neighboring pixels or the intensity value of the pixel in consecutive image frames. Here we enforce spatial and temporal constraints between pixels to determine the degree of dryness of each pixel. We use a **graph-cut** approach to perform the segmentation and to determine the degree of dryness of each pixel. The graph-cut algorithm was described in chapter 4 and it was explained that the cost function to be minimized is defined on a neighborhood $\mathcal{N}$. Therefore, it lets us incorporate constraints between neighboring pixels,$\{i, j\} \in \mathcal{N}$ as described later.

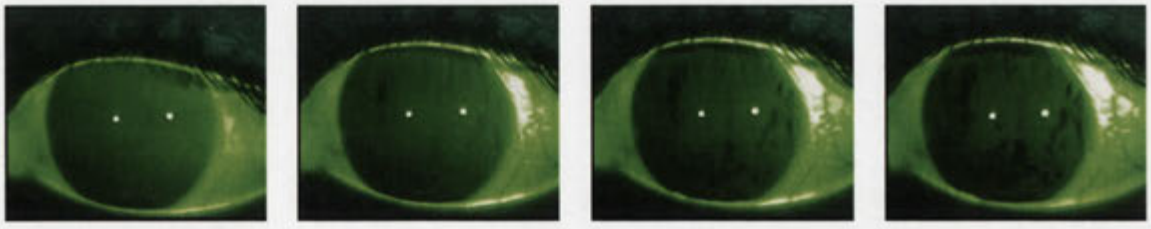


Figure 6.1: *A sequence of images showing how dryness forms over the iris. The first image is immediately after a blink and then the subsequent images are at every 4 seconds. The black areas are related to dryness symptoms. The darker the area, the drier it becomes.*

We transform the aligned dry eye video (see Section 5.4) into a spatio-temporal 3D volume, so a relationship between successive images is defined. The 3D image volume is modelled as a 3-dimensional multi-label Markov Random Field (MRF) in which the label assigned to each pixel represents the degree of dryness. In addition, we introduce the idea of enforcing temporal monotonicity. This reflects the condition that dry spots on the iris can only become darker (dryer) in temporally successive images as seen in Fig. 6.1. To enforce the increasing-dryness condition, we define asymmetric edge weights in the temporal direction, specifying an infinite (or very large) cost to assigning decreasing labels to a pixel in consecutive frames. The associated energy minimization problem is solved using the alpha expansion algorithm [17].

The rest of the chapter is organized as follows: A brief description of the alpha-expansion algorithm; We describe our general approach on how to incorporate temporal monotonic constraints; We demonstrate how monotonic constraints are used for dryness estimation to provide more reliable and robust results; Finally, results on eye images are given.

6.1 Formulation of the problem

We formulate our problem as a second-order MRF . In this approach, each variable i must be assigned a label $\mathbf{x}_i$ from the set of labels $\mathcal{L} = \{0, 1, 2, \ldots, \ell\}$. The most probable labelling $\mathbf{x}^\star$ minimizes the associated energy function:

$$E(\mathbf{x}) = \sum_{i \in \mathcal{P}} E_i(x_i) + \sum_{(i,j) \in \mathcal{N}} E_{ij}(x_i, x_j). \tag{6.1}$$

Here, $\mathcal{P}$ is the set of pixels in the image and $\mathcal{N}$ is the set of pairs of pixels defined over the standard 4-connected neighborhood. The set of labels $\mathcal{L}$ represents the estimated thickness of the tear film. A labelling of 0 represents no thinning of the tear film and the final label ℓ represents a complete absence of fluid, or a break-up of the tear film. Other labels depict the different degrees of thinning of the tear film.

The unary terms E_i are application dependent and we employ a dryness measure similar to the one discussed in the previous chapter in section 5.5.1 and (5.8). The term

should be related to the strength between the intensity of the pixel and the label x_i. The pairwise terms E_{ij} enforce an *a priori* model. In our application, we generally expect the labels of neighboring pixels to be the same (or at least quite similar). In other words, we do not expect to have a fast change from dry pixel to non-dry pixel. However, large changes are also possible at edges. For example, circular break areas can be small and local and have sharp edges (see short discussion in Section 5.5.3). Therefore, we employ a function based on truncated linear distance (see Fig. 6.4(a)), which encourages local smoothness, while limiting the cost of large changes to a threshold T:

$$E_{ij}(x_i, x_j) = \lambda \min(|x_i - x_j|, T). \tag{6.2}$$

In this function, x_i and x_j are the labels assigned to pixels i and j and the energy value is proportional to the difference between the labels, i.e. the difference between the degrees of break. The alpha-expansion algorithm [17] can minimize functions of the form (6.2). The solution is an image, where each pixel is assigned a label depicting its degree of dryness. Although an optimal solution is not guaranteed, in practice the method performs quite well.

6.2 The alpha-expansion algorithm

The alpha-expansion algorithm can be seen as an extension to the original graph-cut algorithm to detect multiple objects in an image. The graph-cut algorithm was presented in section 4.5 and finds the minimum of a function of the form of (6.1) when using only two labels. However, in this chapter we would like to segment multiple labels, i.e the different degrees of thinning. Therefore, the label set is defined as: $\mathcal{L} = \{l_1, \ldots, l_m\}$, where each label can be thought as a separate object. One of the main usages of this algorithm is for depth and motion estimation in stereo images. These applications either use the original alpha-expansion graph-cut algorithm or a variation of it, which consists of a different way to build the graph, such that is suits the specific application or the cost function [12, 63, 66, 77, 114, 123].

The alpha-expansion algorithm attempts to find the optimal labelling by itera-

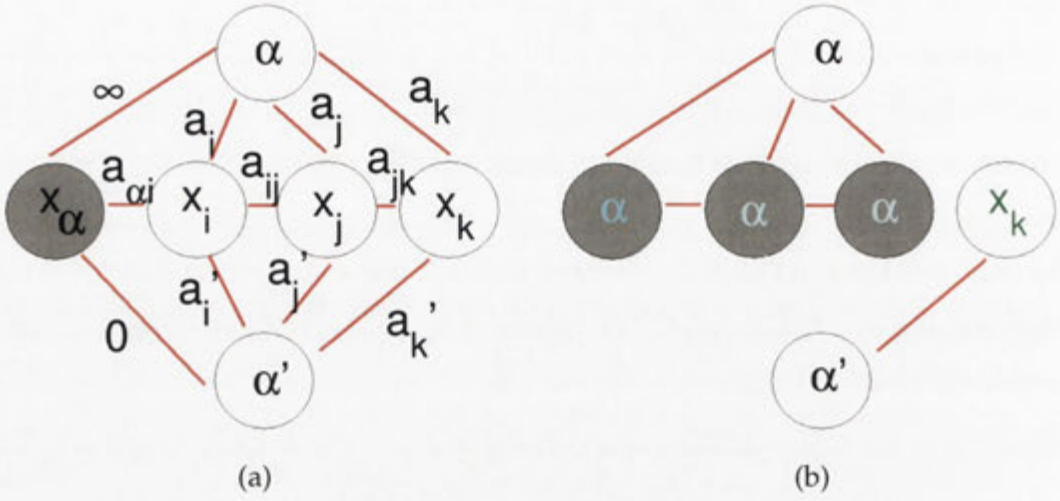


Figure 6.2: *Schematic example of alpha-expansion using 4 pixels. (a) Before the expansion move, the pixel x_α is already labelled as α and the other three pixels have different labels. The edge weights correspond to the unary and pairwise costs of the function. (b) After the expansion move, a cut is produced. The three grey pixels are now segmented as α and x_k has retained its original label as it is not connected to α.*

tively trying expansion moves. At each iteration, the expansion move changes the label of a set of image pixels to α and checks whether the cost is reduced. If it is reduced, the new labelling is used for the next iteration. The process is repeated for all labels $\alpha \in \mathcal{L}$. Once the loop finished iterating through all labels, the process stops if no decrease in cost has been achieved. If lower cost is found, the process continues iterating through the labels again.

The interesting step in the algorithm is the expansion move performed at each iteration to decide which pixels change their label to α and is depicted schematically in Fig. 6.2. At each iteration, a new graph is built to separate the image pixels to two objects labelled as: α and $\bar{\alpha}$ (or equally to object and background). Such a graph includes all image pixels and two terminals labelled as α and $\bar{\alpha}$ and can be seen in Fig. 6.2(a). The edge weights a_i and a_{ij} correspond to the unary and pairwise costs respectively. It can be proved that the exact assignment of the edge weights is the same as in the regular binary case. In the example, we assume that one pixel is already assigned as α and the others have different labels. The cut of such graph can be computed using the min-cut algorithm and thus it defines a labelling. If a pixel in the cut is connected to

α, it changes its label to α. Otherwise, it keeps its previous label. An example of a cut is depicted in Fig. 6.2(b). Two pixels changed their label to α and the cost of the cut is: $a'_i + a'_j + a_k + a_{jk}$. We note that after an expansion move, the set of pixels labelled as α can only grow, so the graph assigns an infinity cost to edges between pixels which are already labelled as α and the terminal α.

The alpha-expansion algorithm does not necessarily produce the optimal solution as not all labellings are examined. Depending on the specific form of E_{ij}, one can solve the problem exactly using Ishikawa's graph construction [53]. Although an exact solution to our problem formulation is possible, the magnitude of the problem makes such an approach infeasible. Ishikawa's algorithm is mainly used to segment small images and not 3D volumes, as we would like to segment in the dry eye problem.

In order for the alpha expansion algorithm to be solvable, a similar inequality to the submodularity requirement of (4.13) has to be satisfied. We remind the reader that the inequality between pixels i and j is: $E_{ij}(0,0) + E_{ij}(1,1) \leq E_{ij}(0,1) + E_{ij}(1,0)$. The terms on the left side state that there is no change of labels between the pixels. The terms on the right side state that there is a change of label between the pixels. Similarly, it can be applied to the alpha-expansion algorithm.

The idea is that at each iteration a binary problem is solved with the variables α and $\bar{\alpha}$ (or similarly 1 and 0). We assume that the original labels of pixels i and j are x_i and x_j respectively. A change of labels happens when a pixel is assigned α (or 1) instead of its previous label. A pixel keeps its label when it is assigned $\bar{\alpha}$ (or 0). We do the following substitution to the original submodularity inequality: $1 = \alpha$ and $0 =$ original label. Therefore, the term $E_{ij}(x_i, x_j)$ means that both pixels keep their original labels and $E_{ij}(x_i, \alpha)$ that i keeps its original label and j changes to α. Then the condition to be satisfied at a specific iteration is:

$$E_{ij}(\mathbf{x}_i, \mathbf{x}_j) + E_{ij}(\alpha, \alpha) \leq E_{ij}(\mathbf{x}_i, \alpha) + E_{ij}(\alpha, \mathbf{x}_j), \forall \mathbf{x}_i, \mathbf{x}_j. \tag{6.3}$$

The left side states that either both pixels keep their labels or both are assigned the same label. The right side states that one pixel changes its label. This condition has to be satisfied at each iteration of the algorithm for any two pixels and any label α.

It is possible to rearrange the costs of (6.1) such that $E_{ij}(x, x) = 0$ for all pixels and all labels x by pushing flow through the graph (see section 4.5). Therefore, the condition to be satisfied becomes the triangle inequality:

$$E_{ij}(\mathbf{x}_i, \mathbf{x}_j) \leq E_{ij}(\mathbf{x}_i, \alpha) + E_{ij}(\alpha, \mathbf{x}_j), \forall i, j \in \mathcal{P}, \forall \mathbf{x}_i, \mathbf{x}_j, \alpha \in \mathcal{L} \tag{6.4}$$

The inequality states that for two neighboring pixels, the cost of a change between the labels x_i and x_j, should be lower than changing via a third label α. We will show that the smoothness constraint of (6.2) or a linear function based on label difference can be solved using the algorithm. In general, any concave function can be solved using the algorithm. However, quadratic functions, such as $(\mathbf{x}_i - \mathbf{x}_j)^2$ or the one used by Veksler in [114], cannot be solved since the triangle inequality does not hold.

A commonly used function that satisfies (6.4) is the Potts model [17]:

$$E_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 0 & \text{if } \mathbf{x}_i = \mathbf{x}_j \\ \gamma & \text{otherwise,} \end{cases} \tag{6.5}$$

where $\gamma > 0$ is some constant. Any difference between the labels of the pixels is penalized equally. Such a model is less suitable for dryness detection, as changes usually happen gradually and sharp change in labels should be penalized more.

As in practice, the expansion algorithm provides good approximation to the optimum and performs relatively fast, it is preferable to Ishikawa's method when it can be used. Furthermore, it has been shown that the expansion algorithm is most suitable for applications that use a pairwise term based on the Potts model or a linear difference of labels.

6.3 Previous work

The original paper [17] on alpha-expansion stated that the pairwise cost functions has to be metric (in order to simplify the presentation), and hence symmetric. However, they also mention in a footnote that symmetry is not essential, but still there has been little work on asymmetric cost functions . In a very recent paper [21], a general

method for constructing and solving asymmetric cost functions using graph-cuts is provided. The paper addresses the problem of stereo depth estimation and incorporates a monotonic constraint which is related to the spatial location of the pixel in the image. Objects that appear towards the top of the image are usually further away than objects that appear towards the bottom. They show how such monotonic constraints can be incorporated into the pairwise term of the energy function and that the alpha-expansion algorithm can solve any concave pairwise function.

3D medical image segmentation using graph-cuts have become quite popular. Recently Bokyov *et al.* [15] described a global N-D graph-cut segmentation approach that can be used to segment the kidney from a 3D MRI. They are interested in identifying three regions of the kidney and conduct three independent binary segmentations sequentially. However, they do not employ a multi-label approach to segment all three regions simultaneously.

Another example for an application that uses 3D volume binary graph-cuts is for the segmentation of brain tumors [119]. The input images are 3D MRIs and the segmentation is performed over the whole volume using the standard 6-neighborhood. The unary term is application dependent and is based on learning of features. The pairwise term is the same one both spatially and for voxels in different slices and is based on the voxel's intensity and distance from each other.

Grady and Jolly [36] study different graph topologies and weighting functions to segment a set of 62 3D CT images. Their experiments include the standard 6-connected, 26-connected neighborhoods and also a 10-connected, which is a combination of 8-connected in the XY-plane and 6-connected for the Z-plane. The pairwise cost functions (or weighting functions) include 3 different functions based on intensity difference between two voxels: Gaussian, Reciprocal and Histogram based. In the latter function, the cost (weight) is proportional to the probability of the voxel's intensity being foreground and it is computed using seed points that were selected manually or automatically.

In their experiments, the manually segmented area of the 3D volumes was compared with the graph-cut result. The results show that the histogram based method is superior to the other weighting functions. Then, if histogram based model cannot

be established, the reciprocal weighting function should be used. In terms of the best topology, the 26-connected performed the best and surprisingly the 6-connected outperformed the 10-connected. These results demonstrate the advantage of one weighting function over the other. However, in our case, we segment multiple objects simultaneously. Our unary term is based on intensity values, but the pairwise term is a function of label difference. Using intensity values for the pairwise term as well (as in their experiments) puts more emphasis on the pixel's intensity. Employing a 26-connected neighborhood can be beneficial, but it increases the complexity of the problem, especially in the multi-label case.

Asymmetric cost functions have not gained widespread use. In [121], the authors employed the alpha-expansion for spatial geometric constraints. The asymmetry in their algorithm is used to distinguish between the relative layout of objects using terms such as above/below/left/right. Their cost function, which is set through a learning process, is submodular in most cases (over 99% of the cases) and therefore concave. However, according to the authors, the alpha-expansion algorithm was not able to find a good solution and converged to a local minimum. To overcome this problem, they used an annealing algorithm, where the pairwise potential is weakened. One possible reason that the alpha-expansion algorithm did not produce good results is the complication of their pairwise and unary functions.

In [79], Liu et al present a graph-cut based algorithm to segment an image to multiple labels. Their label set is defined as: left,right,center,top and bottom. Relationships between these regions are defined and the corresponding constraints, such as that a pixel labelled left cannot be to the right of a pixel labelled right. In addition if a pixel is labelled as a center its neighbors can only be labelled according to their spatial location. The pairwise function is based on Potts model and incorporates these constraints by assigning an infinity cost if they are violated. In addition, the function is asymmetric such that the cost of labelling neighboring pixels as 'right' and 'center' is not the same as assigning 'center' and 'right'. The suggested solution is a variation of the alpha-expansion. They define horizontal and vertical moves and at each iteration, a pixel can change its label to a set of three labels. This is different from the alpha-expansion algorithm where a pixel can change its label only to α.

The authors state that the alpha-expansion algorithm will usually not find the global minimum with such constraints. However, we note that the constraints in the paper are two-dimensional. The monotonic constraint we will discuss later is defined only in one direction (temporally). Therefore, the alpha-expansion moves result in a lower energy cost throughout the iterations, as they do not depend on a number of labels being changed simultaneously in order to obtain a valid segmentation. This is not the case described in [79], where an assignment with a lower energy cost usually requires change more than one label.

In [59], the authors use an asymmetric cost to segment multiple surfaces in 3D CT images. Even though the surfaces are segmented simultaneously, they use a binary label set (and not a multi-label approach). To our knowledge, asymmetry has not been used before to enforce temporal constraints within volumetric images.

6.4 Our approach

6.4.1 3D graph construction

Graph-cuts minimization is not limited to 2D applications and can be extended easily to 3D applications. The main advantage of a 3D approach to segmenting individual 2D slices is that the relationship between pixels at consecutive slices is considered. Moreover, it allows one to incorporate monotonic constraints (described in the next section) between slices, which would have been impossible otherwise.

Extending the 2D approach to 3D is based on redefining the neighborhood $\mathcal{N}$ used in the pairwise term of (6.1). While in the case of MRI segmentation, it is fairly clear what the individual slices are, we offer an approach based on temporal progress. Even though the image modality is 2D in the case of the FBUT test, it can be perceived as a 3D approach to capture the global relationship between image frames. Denoting image t in a video of length $n+1$ as I_t, each image is considered as a horizontal slice in the 3D graph (or MRF), creating a graph based on temporal changes. Therefore, slice number 0 in the graph is the image immediately after the blink and slice n is the last image in the sequence. Every other slice is related to the time passed since the blink.

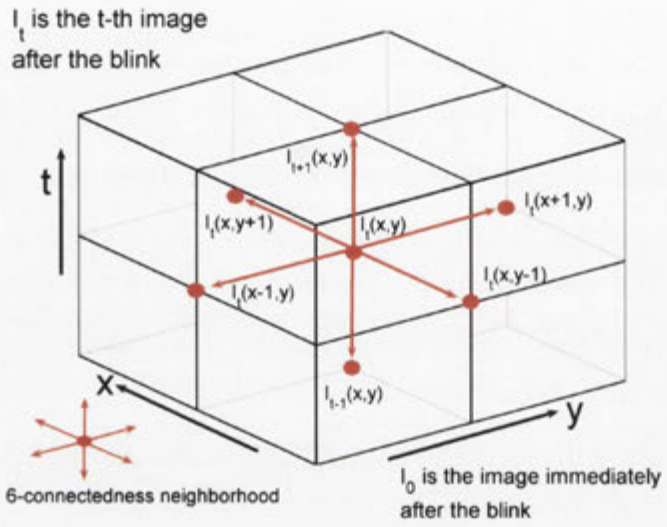


Figure 6.3: *Explaining the 3D MRF. The x and y axis are the image planes; the t plane is the temporal plane. A pixel $I_t(x, y)$ (or the voxel (x,y,t)) is the pixel (x,y) in the t-th image after the blink. The figure shows the 6-connected approach (by the red arrows), where a pixel is connected to its 4 immediate neighbors in the same plane and to two pixels corresponding to the same location at times $t-1$ and $t+1$. Each voxel in the MRF can be assigned any label from the set $\mathcal{L}$.*

The construction of the graph is based on 6-connected neighborhood $\mathcal{N}$, and an example of a 3D MRF showing the 6-connected neighborhood is depicted in Fig. 6.3. Each voxel in the MRF $(x, y, t), t = \{0, \ldots n\}$ is connected to its four immediate spatial neighbors in the same image (except for pixels in the image borders) and to the corresponding pixel (x, y) in the previous and next frames: $(x, y, t-1)$ and $(x, y, t+1)$ (except for the first and last frames). Another way to look at the neighborhood of a voxel is: $\mathcal{N} = \{left, right, up, down, next, previous\}$. Therefore the energy function is still built only from quadratic terms, as each voxel can be seen as being part of a maximum of 6 pairwise cliques. Each voxel is now also dependent on two voxels which are temporally different. This allows the addition of time based constraints. Denoting the set of pixels of frame t by $\mathcal{P}_t$, the new set of pixels is defined over the whole image sequence: $\mathcal{P} = P_0 \cup \ldots \cup P_n$. The hidden nodes of the MRF are the labels assigned to each voxel from the set $\mathcal{L}$.

6.4.2 Monotonic constraint

Multi-label problems usually have an inherent meaning to the ordering of the labels. In our case, the labels depict different degrees of break progressing through time. Hence we expect the labels to have a gradual temporal change. So we employ a distance metric for the pairwise cost in the *temporal* axis:

$$E_{ij}(x_i, x_j) = \gamma |x_i - x_j|. \tag{6.6}$$

As long as the patient does not blink, the thickness of the tear film cannot increase between consecutive images. Formally, the label x_j of a particular pixel at time t_j must be less or equal to the corresponding label x_i of the same pixel at time $t_i = t_j + 1$. We enforce this monotonic dryness condition directly into the pairwise energy term:

$$E_{ij}(x_i, x_j) = \begin{cases} \infty & \text{if } t_i = t_j + 1 \text{ and } x_i < x_j \\ f_{ij}(|x_i - x_j|) & \text{otherwise} \end{cases} \tag{6.7}$$

(In this equation we assume without loss of generality that $t_i \geq t_j$). The generic function $f_{ij}(|x_i - x_j|)$ in (6.7) refers to either (6.2) or (6.6) depending on whether i and j are a spatial ($t_i = t_j$) or temporal ($t_i = t_j + 1$) pair.

The monotonic function (6.7) sets an infinite cost to any labelling $\mathbf{x}$ where a pair of labels (x_i, x_j) for a particular pixel at times t_j and $t_i = t_j + 1$ decreases — i.e., $x_i < x_j$. Although we associate an infinite cost for violating monotonicity, in general, a finite cost can be employed.

Fig. 6.4 shows two examples of pairwise functions which can be minimized using alpha-expansion. Part (a) is a cost function based on (6.2). The maximum penalty for assigning different labels is bounded by T. Part (b) is a cost function based on (6.6). If the change of labels is negative, the cost is infinity; Otherwise, the penalty is linear and not truncated. When $x_i = x_j$ the function is assigned 0. However, it is not mandatory. In our algorithm, we use the first function for spatially neighboring voxels as a large change between labels might happen at the edges of break areas. The second function is used for temporally neighboring pixels, where changes in labels

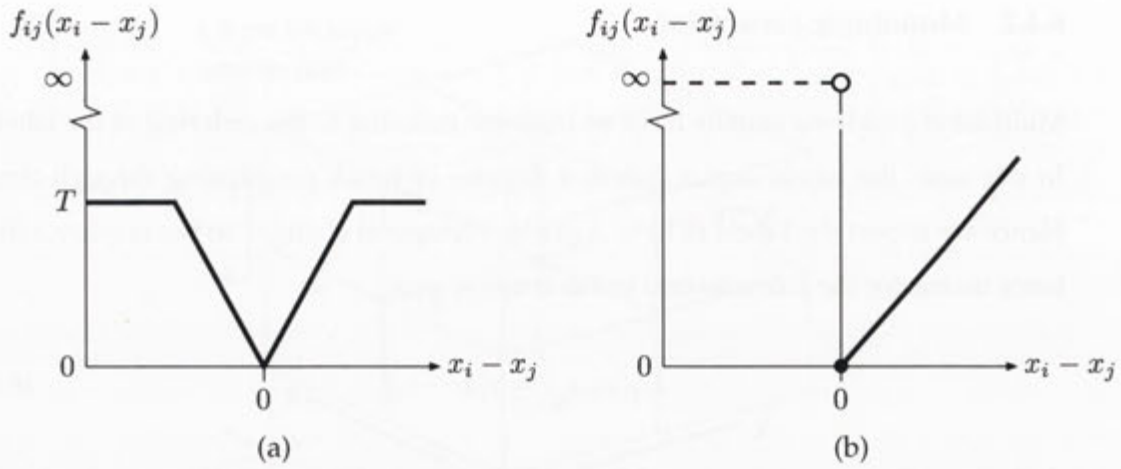


Figure 6.4: *The spatial (a) and temporal (b) pairwise functions. The graphs are presented as a function of the difference between the labels x_i and x_j. (a) Neighboring pixels within the same frame are encouraged to have the same label unless the difference is quite big. In this case, there is no preference between a medium and large difference. (b) Temporally, labels must not decrease as time progresses. Moreover, the increase (if any) should not be too large.*

(dryness) are usually gradual through time.

The inclusion of a monotonic constraint makes the pairwise terms asymmetric: the cost of changing from label α to β can be different from changing from β to α, or mathematically $E_{ij}(\alpha, \beta) \neq E_{ij}(\beta, \alpha)$. In order to solve the asymmetric construction using alpha-expansion, the triangle inequality of (6.4) has to be satisfied. First, we will show that it is satisfied when excluding the monotonic constraint. Thus, for the linear penalty of (6.6), the following has to be satisfied for every x_i, x_j and α:

$$|x_i - x_j| \leq |x_i - \alpha| + |\alpha - x_j|. \tag{6.8}$$

If $x_i \leq \alpha \leq x_j$ or $x_j \leq \alpha \leq x_i$, the terms at each side of the inequality are the same. In all other cases, the inequality clearly holds as either $|x_i - \alpha|$ or $|\alpha - x_j|$ is greater than $|x_i - x_j|$. When using the truncated linear penalty of (6.2), the following has to be satisfied:

$$\min(|x_i - x_j|, T) \leq \min(|x_i - \alpha|, T) + \min(|\alpha - x_j|, T). \tag{6.9}$$

If any of the terms in the right side equals to T, then it clearly holds. If the term on the

left equals to T ($|x_i - x_j| \geq T$) and neither of the terms on the right side equals to T, we know from (6.8) that the sum of the the right side is greater or equal to $|x_i - x_j|$. But this has to be greater than T so the inequality holds. In all other cases, the truncated linear function behaves the same as the linear penalty function. Finally, we have to consider the monotonic constraint when $t_i = t_j + 1$ and $x_i < x_j$:

$$\infty \leq E_{ij}(x_i, \alpha) + E_{ij}(\alpha, x_j) \tag{6.10}$$

The inequality will not hold if both terms on the right side are finite. According to (6.7), this could happen only if α satisfies the following: $x_i \geq \alpha \geq x_j$. However, since $x_i < x_j$, there is no such α, so at least one of the terms on the right side equals to infinity.

6.4.3 Summary of approach

Before presenting the actual details of the implementation, we summarize the steps in our proposed segmentation approach:

1. Obtain a series of images of the same object. In the case of a dry eye video, images of the anterior of the eye are taken throughout time.

2. Align the images according to the object of interest (if needed). For dry eye images, we align the iris to be at the same location and size at all images.

3. Define the relationship (neighborhood) between individual images (slices) to create a 3D graph.

4. Define the number of labels needed for the segmentation. The number of labels defines how fine the segmentation is and in the case of dry eye, distinguishes between different degrees of thinning of the tear film. Although large number of labels may produce finer results, it requires longer computation time.

5. Define the cost function for the unary and pairwise terms. It is possible to have a different pairwise function spatially and between slices.

6. Define the monotonic constraint and revise the pairwise term. For dry eyes, the constraint is defined temporally - the amount of fluid in the eye since the blink (and before the next blink) cannot increase.

7. Minimize graph and produce a labelling for each voxel.

6.5 Application to detect dryness

The 2D segmentation approach of chapter 5 first detects the iris in each of the video frames. Then, the images are aligned according to the iris to create a new video, such that the iris is located roughly at the same location at each image. Finally, the dry spots are segmented based on analyzing the aligned video. The cost function of (5.8) examines differences in intensities for each of the pixels of the iris between the first and last images in the video. A dryness image is created, where each pixel is assigned an intensity value which is proportional to its degree of dryness and is denoted by $\tilde{I}(x, y)$. However, the degree of dryness is also computed at each individual slice, and we denote this pixel value by $\tilde{I}(x, y, t)$. The DEBUT is detected by scanning the whole video and looking for the first frame when an area of a certain size is over the minimum intensity considered for a break (see (5.10) and (5.12)).

This approach produces good segmentation results and is very fast. Nevertheless, it has a few objective disadvantages:

1. Small errors in the alignment can bias the dryness result for a pixel. This is especially noticeable near the eyelids and the borders of the iris with the conjunctiva.

2. The spatial relationship between neighboring pixels in the 2D image is not considered. Therefore, there is no smoothing and the pixel's degree of break is unrelated to its immediate neighbors.

3. There is no use of knowledge regarding the temporal change.

We will now show how these issues can be resolved by employing our 3D monotonic graph cut approach.

6.5.1 Advantages of the 3D approach

Given the aligned video created by the 2D segmentation approach, it is possible to incorporate the ideas discussed so far to improve the segmentation results. Instead of looking at individual pixels and examining every single 2D image for the Break Up Time (BUT), we add the following assumptions:

1. Smoothness constraint - If a pixel becomes dry, it is likely that its neighbors also show a certain degree of dryness. Similarly, if a pixel has no dryness, it is likely that its neighbors are not dry as well.

2. Using temporal knowledge - The video is considered as a 3D volume where each 2D frame is a slice in the 3D image. Therefore, the 3D volume is built of temporally successive images (representing temporal progress of dryness) stacked on top of each other. Segmenting the 3D volume takes into consideration the relationship between the pixel's values at all times.

3. Monotonicity constraint - Temporally, pixels should only become darker, as the amount of fluid in the tear film decreases as time passes. If a pixel becomes brighter, it is probably caused by an error in the alignment process or because of shifting of the fluorescein after the blink and not related to the actual dryness.

6.5.2 Applying the technique

Referring back to section 6.4.3, we show now how the described approach can be easily adopted to the dryness problem. Given the aligned video created in the 2D approach, it is used to create a 3D graph based on temporal changes (see Sec. 6.4.1). The region of interest in each image is defined as only the pixels belonging to the iris. This region should not be image dependent as after the alignment, the iris is resized to the same size at the same location.

The number of labels needed for segmenting dryness depends on the importance of distinguishing between different degrees of thinning of the tear film. A reasonable choice is to use a set of 9 labels: $\mathcal{L} = \{0, 1, \ldots, 8\}$. This number of labels generally produces suitably precise segmentations of the tear film.

The unary term is defined using the dryness value $\tilde{I}(x, y, t)$, which we showed how to compute in section 5.5.1. This value is computed for every pixel for every image $t = \{0, \dots n\}$ in order to evaluate the DEBUT. When using a multi-label algorithm, a value has to be assigned for each label $E_i(\mathbf{x}_i), x_i \in \mathcal{L}$. The value $\tilde{I}$ can be associated with the expected label x_i^* for each pixel. For example, the intensity range of $\tilde{I}$ can be divided into $|\mathcal{L}|$ equally spaced bins, where each bin is associated with a label. The unary term is then defined as a function h proportional to the difference from the expected label:

$$E_i(x_i) = [h(x_i - x_i^*)]^2. \tag{6.11}$$

This value is supposed to be higher than the pre-defined threshold B_T when the pixel has a break and a value of 0 when there is no break. We employ a simple model based on a Gaussian distribution with a σ that is related to the uncertainty in $\tilde{I}_t(x, y, t)$. We first define the bin size b and the center of each bin m_i which is associated with label x_i:

$$b = (B_T - 0)/(|\mathcal{L}| - 2),\ m_i = (x_i * b) - (b/2). \tag{6.12}$$

The cost of assigning label x_i to voxel $i = (x, y, t)$ is defined as a function of the pixel's intensity in the dryness image and the label associated with the bin's center:

$$E_i(x_i) = P(x_i|\tilde{I}_t(i)) = \mathcal{N}(\tilde{I}_t(i); m_i, \sigma). \tag{6.13}$$

The pairwise term (6.7) uses linear distance metrics in both the spatial (6.2) and temporal (6.6) directions with parameters λ and γ. This metrics penalizes linearly for changes in labels both temporally and spatially, where $t_i = t_j$, since both pixels belong to the same image. The value of these parameters depends on the number of labels in $\mathcal{L}$. In the spatial case the function is truncated, since large label discontinuities can happen as some break-up areas are local in shape and they should not be over-penalized. However, since the number of labels in $\mathcal{L}$ is not high and label assignment is expected to be smooth in most regions, λ should be high.

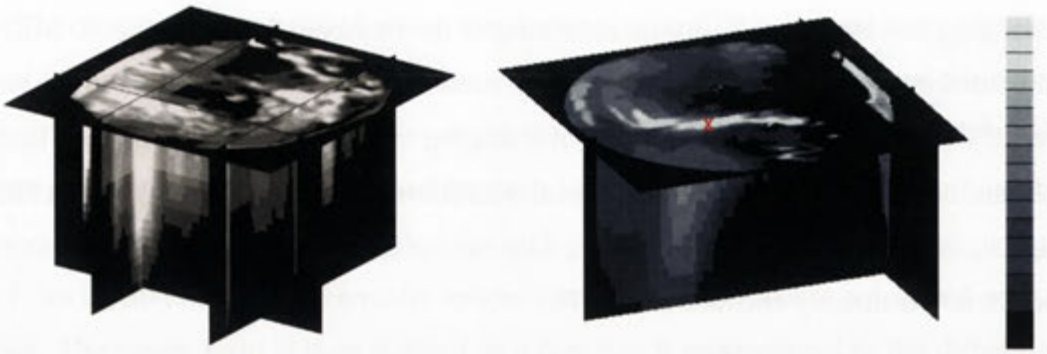
In the temporal domain, large discontinuities are not expected, so the regular linear distance metric is appropriate. The value of γ is related to the rate of temporally

changing labels and is also tuned according to the number of slices in the 3D MRF. If the number of slices is high, label changes occur slowly and γ should be high. For a small number of labels in $\mathcal{L}$, the rate of changing labels is expected to be low, thus γ should be high. Finally, since the clinical definition of tear film thinning in the FBUT test states that the thickness of the tear film cannot increase with time, the monotonic restriction is directly encoded into (6.7).

Despite its simplicity, the pairwise term is quite appropriate in most cases. A different value for λ might be needed depending on the ratio between the number of slices n and labels $|\mathcal{L}|$ and how fast a change from the minimum to the maximum label can happen. For example, if there is a small number of slices, it is possible to have a faster change between labels, so a lower value might be assigned to γ.

This finalizes the steps required for the creation of the graph. The minimum cut is found by using the alpha-expansion algorithm. The labelling for each voxel is its degree of dryness at the time. The labelling of the pixels at time n should be similar to the value of the same pixel in the dryness image $\tilde{I}$ computed by the 2D approach (see section 5.5.1).

The DEBUT is also computed in a similar way to (5.10) of the 2D approach. The slices in the volume are scanned from bottom (time $t = 0$) to top. We would like to find the first slice such there exists a break region. In every slice, we are looking for a connected region labelled as $m - 1$ (the label associated with a break). Each connected region's size is compared with the threshold of (5.12). If it is over the threshold, the time passed since the blink corresponding to this slice is assigned as the DEBUT. The monotonicity constraint guarantees that the number of pixels in a break region will not decrease in succeeding slices. Therefore, it is impossible for a pixel to not have a break or show further breaks after it reaches its final label of $m - 1$. This is not the case in the 2D approach, where theoretically a pixel can have a computed dryness intensity of $\tilde{I}_t(i) > B_T$ at time t and then at later times: $t_k > \ldots > t_2 > t$ have increasingly higher dryness intensities: $\tilde{I}_{t_k}(i) > \ldots > \tilde{I}_{t_2}(i) > \tilde{I}_t(i)$.



(a) (b) (c)

Figure 6.5: *The top slice shows the final segmentation result. The brighter the colors, the more severe the dryness. The temporal axis (or the t-axis) is presented from bottom to top of the image. Thus, the image at the top of the figures is the dryness image just before the next blink. The 3D view demonstrates how the dryness progresses at specific spatial locations. The monotonic constraint forces pixels to have non-decreasing intensity. (a) Dryness image of the sequence from Fig. 6.1 (rotated counterclockwise for clarity). (b) A sequence where the dryness is mainly in the central and superior parts (rotated clockwise). (c) Temporal progress of the voxel highlighted by* $\mathbf{x}$ *in (b)*

6.6 Results

In order to test our method, we used a database of 22 videos with varying length and degrees of thinning of the tear film. The iris was aligned in each of the videos as described in section 5.4. The aligned videos had up to 140 image frames. However, in order for the alpha-expansion algorithm to run in a reasonable time, we only used every 5th image, therefore limiting the number of images to 28. Such reduction in the number of images is reasonable for producing the dryness image as the changes are usually very gradual. In order to detect the exact frame of the DEBUT, another search can be done in a small number of frames.

We produce a 3D volume image showing the progress of the dryness through time and a separate 2D image of the top slice, which is similar to the output given by the 2D approach. Fig. 6.5(a) shows the result for the sequence in Fig. 6.1. The brightest pixels correspond to areas of maximum thinning. The top slice is the final segmentation result. The temporal axis (bottom to top of the volume) shows the progress of dryness through time. It can be seen how the monotonic constraint enforces the voxels to have only a non-decreasing intensity (see Fig. 6.5(c)) and that some of the voxels start

showing dryness at a later stage but progress faster. Fig. 6.5(b) shows a similar cut for a different sequence where dryness mainly develops in the central and superior areas. The area of dryness in the superior part is quite thin in size, but the smoothness constraint ensures it is a connected area.

In order to show the contribution of the monotonic constraint, we examined the average number of labels changed between every two consecutive slices:

$$\mathcal{C} = 1/|\mathcal{P}_i| \sum_{x \in \mathcal{P}_i} \sum_{t=0}^{t=n-1} |x_i^{t+1} - x_i^t|. \tag{6.14}$$

We denote the label of pixel i at time t by x_i^t. The sum of (6.14) is normalized by $|\mathcal{P}_i|$ which is the number of the pixels in the iris. However, we do not normalize by the the video's length, since all videos in our experiment have a break at some time. The measurement we are interested in is just the number of label changes and not the time required to reach the break.

When using the monotonic constraint, the upper bound for $\mathcal{C}$ is defined by the max number of labels: $\mathcal{C} \leq \ell$ (or 8 in our experiments). This is since the monotonic constraint forces each pixel to have only increasing labels and the maximum label number is 8. In the 2D approach we do not have such a constraint, so labels does not have only to increase. In order to be able to use (6.14) on the 2D approach, we converted the dryness values of all pixels at all times to a label between 0 and ℓ. We expect to have a lower number of changes in the 3D approach when compared to the 2D approach. A lower number should mean that the segmentation is more robust and pixels do not change their dryness value too often, which would have impact on the computation of the DEBUT. We note that a higher number of changes per pixel in one video compared to the other does not necessarily mean that the result is worse, but possibly that larger areas became gradually dry. However, when using (6.14) on the same video to compare between two approaches, a lower number of changes is meaningful, since the size of the dryness area is equal.

The result of applying (6.14) to both methods on all 22 videos are summarized in Table 6.1. We received an average of 2.72 and 1.55 for the 2D approach and 3D approach respectively. The last column of the table shows the percentage of iris pixels

that had at least one label change. In a few videos, more than two thirds of the iris pixels had no label change, so the changes are focused on a small subset of iris pixels. Therefore, the difference in $\mathcal{C}$ between the approaches is more meaningful. In the 2D approach, individual pixels mainly near the eyelids or the iris's borders, had up to 68(!) label changes, while a maximum of 8 is enforced when using the monotonic 3D approach.

Video number 20 had the highest ratio of label changes between the two approaches. The 2D approach struggled with the quick movement of the iris and the eyelids that happens after around 240 frames. It takes a few iterations for the alignment algorithm to detect the iris again. As a result, the dryness images contain big areas that become suddenly dry and then non-dry after the alignment is corrected. In the 3D approach, where a monotonic constraint is enforced, these areas do not become dry at all through this transition. This is depicted in Fig. 6.6, showing 6 consecutive dryness images produced by the new approach and the corresponding dryness images when using the 2D approach. Even though the iris is temporarily not detected properly, the dryness images are robust. Due to the monotonic constraint, the areas at the left side of the iris are not segmented. Therefore, wet areas are not being marked as dry. If they were marked as such, later on, once the iris is tracked correctly again, they would have to decrease in label (back to wet areas), which is not permitted. This is not the situation in the 2D approach, where wet areas were marked as dry between times $t = 240$ and $t = 280$. Finally, we note that the final dryness image of the two approaches is quite similar. The new approach is a bit more conservative in segmenting pixels (as seen in the figure), but can be adjusted by changing the values used to compute the unary term, for example in (6.12).

Fig. 6.7(a) shows another segmentation result using our approach. Parts (b) & (c) are temporal cuts, where the y-axis in these images is progression through time. Notice how near the left end side the monotonic constraint creates a smooth transition between labels with no fluctuations while there is a lot of noise in the other approach.

Generally, our approach tends to minimize the number of incorrectly segmented pixels when compared to the 2D approach, enforced by the monotonicity. This is useful as the DEBUT is computed as a function of the time passed since the blink

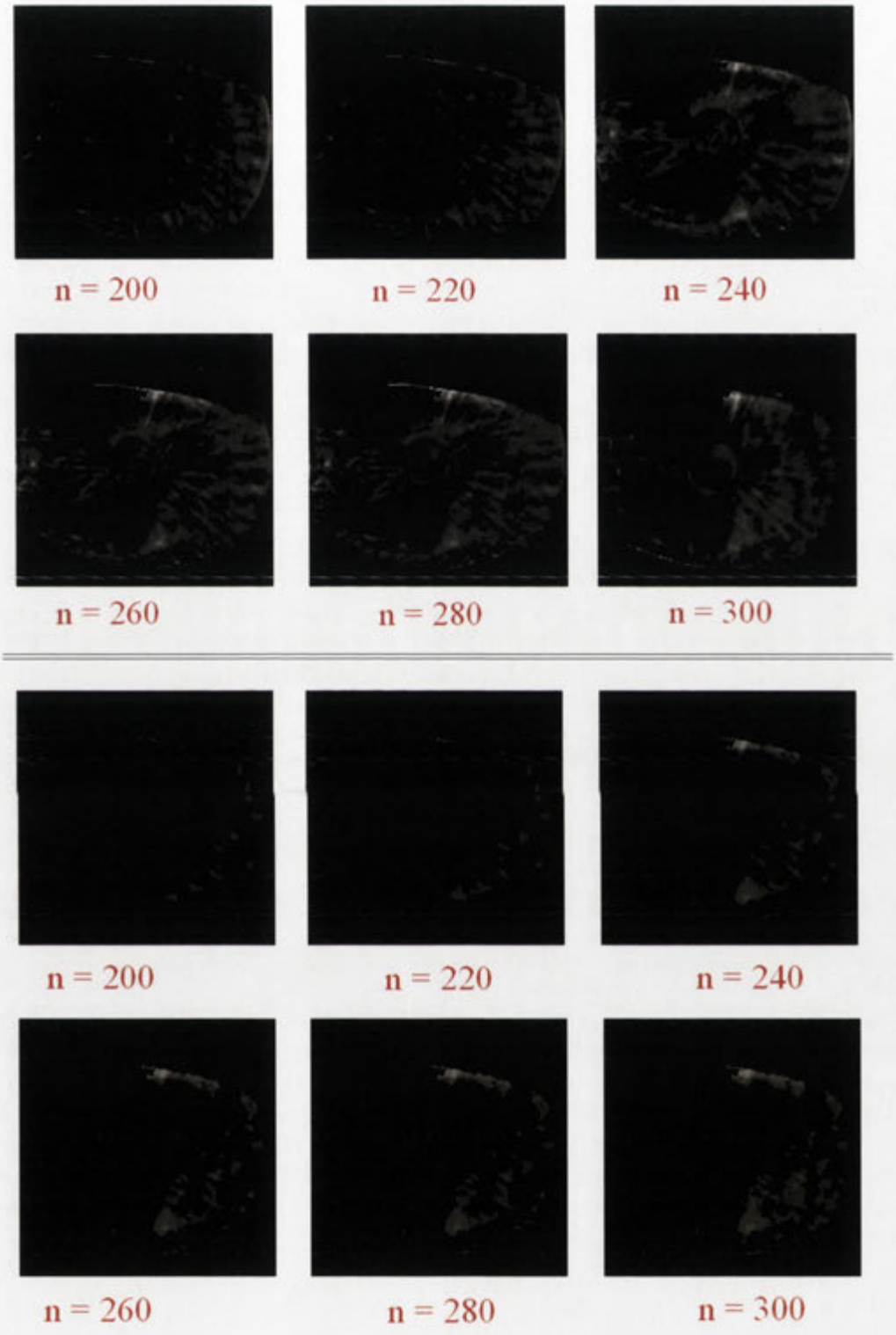


Figure 6.6: *Demonstrating the robustness of the new approach. Six dryness images are shown, each one 20 frames apart from the next one. As in the monotonic approach, we do not use all video frames, the displayed images are continuous slices in the resulting image volume (thus every 20 frames, a slice is added to the volume).* ***Upper part****: Dryness images when using the 2D approach.* ***Lower part:*** *Dryness images when using the 3D approach. We notice the increase in the segmented area in the left side between frames 240 and 280 in the original approach. Then these segmented regions mostly disappear at around frame number 300. This does not happen in the monotonic approach, as it is less sensitive to the temporary misalignment.*

Video No.	$\mathcal{C}$ - 2D approach	$\mathcal{C}$ - 3D approach	max changes	% of changed pixels
1	1.1286	0.5387	30.0000	0.2538
2	4.5329	2.6412	43.0000	0.6688
3	1.1676	0.4774	**68.0000**	0.2187
4	1.2132	0.7522	40.0000	0.4383
5	4.1809	2.8993	35.0000	0.7799
6	2.0526	1.1882	46.0000	0.5765
7	3.3598	1.3768	31.0000	0.7528
8	2.5279	1.4984	31.0000	0.4685
9	**6.4223**	**5.3399**	37.0000	**0.9184**
10	1.7694	0.4003	35.0000	0.3038
11	**0.2649**	**0.1029**	27.0000	**0.0486**
12	2.7301	1.7559	49.0000	0.6871
13	1.4124	0.5122	34.0000	0.2959
14	0.6711	0.1960	**23.0000**	0.1530
15	2.2193	0.9784	36.0000	0.2590
16	3.2871	1.5051	57.0000	0.6441
17	5.1117	3.0674	31.0000	0.8494
18	1.7002	0.4519	29.0000	0.3549
19	2.3577	1.3973	28.0000	0.4495
20	*3.8590*	*0.5321*	56.0000	0.6993
21	4.7781	3.9088	47.0000	0.7948
22	3.2122	2.6509	44.0000	0.8056
avg	2.7254	1.5532	38.9545	0.5191

Table 6.1: *Summarizing the results of the 3D algorithm compared to the 2D algorithm. The 'max changes' column is the maximum changes of a label for a pixel in the 2D approach. The number of changes in the 3D approach is bounded by 8, and is always lower than the number of changes in the 2D approach. The last column is the percentage of iris pixels that had at least one change of labels. Therefore, a small percentage means that only a small area had any degree of thinning, while a large percentage means that the thinning is spread out. In videos with a high percentage of pixels having any degree of thinning, the difference are smaller. The bold numbers indicate the maximum and minimum in each category. The italic number indicate the video with the highest ratio of changes between the 2D and 3D approaches.*

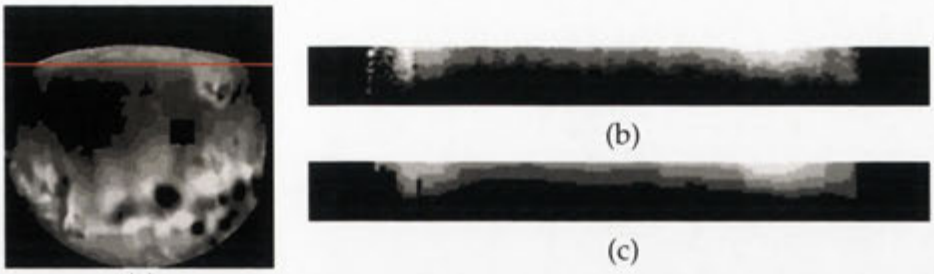


Figure 6.7: *Example of an x-t cut. (a) Segmentation result using the new approach. An x-t cut (not to scale) at the location of the line in (a) using the (b) 2D approach. (c) 3D approach. The y-axis in (b) & (c) shows temporal progress from bottom to top.*

that the size of a certain break area is over a threshold A_{DEBUT} (see section 5.5.2). A low threshold will make the DEBUT detection more sensitive to small breaks that do not progress in size. However, it can result in incorrectly computing a very early DEBUT by the 2D approach, as even a small number of wrongly computed pixels may be enough to be above the threshold. In the new approach, such outliers are less likely to happen, as misaligned voxels cannot temporarily decrease in label and therefore should not have any increase in labels at all. The importance of defining the threshold as a small value is demonstrated in the experimental tests of chapter 7 and, in particular, in section 7.1.1.

6.7 Conclusion and further research

In this chapter, we demonstrated a general approach to create a 3D volume from a video of 2D images and incorporate a monotonic constraint in a graph-cut based solution. This creates an asymmetric graph-cuts problem, which we showed can be solved using the alpha-expansion algorithm. Then, we explained how the approach using the monotonic constraint is applied to dryness segmentation.

The inclusion of a temporally monotonic constraint improves the robustness of the results and reduces the sensitivity to outliers. We also discussed how the computation of the DEBUT can benefit from the monotonic constraint. As outliers will not be segmented unless they persist, the minimum DEBUT area can be decreased to better

correspond to the needs of the clinicians. This is demonstrated in the experimental results chapter at section 7.1.1.

The approach presented in this paper can be extended to other medical applications. For instance, OCT imaging (see chapter 2.3.5) of the retina can provide a number of linear radial scans. An application will try to segment the fixed number of intra-retinal layers to detect an ocular disease around the macular region or the optic disk. Similar steps to those mentioned in section 6.4.3 can be applied to OCT images:

- The 3D volume is created from the six 2D scans taken by the device.
- The images are aligned according to the orientation of the radial scans.
- The neighborhood can be defined as the usual 6-connected.
- The number of labels is equal to the number of layers in the OCT image and is usually assigned to 5 labels.
- The monotonicity is enforced on the y-axis, since the retinal layers always appear on top of each other in the image. Therefore, when scanning the y-axis, the layers have to appear in the known order. In this case, the monotonicity is enforced spatially so the segmentation of layers follows that ordering. In addition, if we know that all layers have to appear at each column, the monotonicity can enforce that label changes should always differ by one. Such a constraint can easily be incorporated into a pairwise term based on difference of labels.

In fluoroscopic imaging involving perfusion of contrast agents, temporally monotonic increase and subsequent decrease of intensity may be enforced using an extension of these methods. Finally, spatial relationships and geometric properties such as convexity may be modelled using MRFs with asymmetric edge labels.

Experimental Results

Through our experiments, we aimed to achieve the following goals:

1. Show that our automatic analysis of the videos is in agreement with the manual analysis done by clinicians. We compare the DEBUT value with the manual BUT, the location of the break and whether there is a meniscus induced dryness.

2. Show that there is a large variation in the BUT value computed by the clinicians and the need for a robust, non-operator dependent value.

In total, we have captured around 200 videos of more than 50 patients. The patients were of different age groups and varied from having a very dry eye (dryness occurring quickly and in several locations) to no dryness at all (no break areas). The clinical routine we have used is based on the FBUT test and is described in section 2.3.4. The experiments demonstrate the performance of the proposed algorithm and explain the importance of defining the DEBUT. In all experiments, the values provided by the clinicians were considered to be the gold standard. Results presented in the form of $x \pm y$ should be interpreted as x is the average value and y is the standard deviation, and all values are given in seconds.

7.1 Defining the experiment

One of the problems with measuring only the BUT by clinicians is the difference in the way each clinician defines the exact time of break. Some clinicians can interpret thinning as a break while others will wait longer to define it as the time of the break. It is also visually hard for the clinician to decide when an area looks black or very black,

especially if the change is gradual. The clinician might focus on a specific region and miss breaks which have faster progress in a different region. For example, an area of thinning in the inferior part might start to form early on. However, another break area in the superior part can have quicker progress even if it starts forming later. In such situations, the clinician needs to keep scanning the iris area to estimate the correct BUT. This is not an easy task as the clinicians tend to focus on a specific part of the cornea (normally the center), making it hard to detect changes near the eyelids.

In order to further corroborate our claim, we have carried out the following experiment to get an idea about the inter-observer (comparing results *between* clinicians) and intra-observer (comparing different estimations of the *same* clinician) variances. A set of 22 videos was chosen with each one of the videos having a full thickness break (a break). The BUTs in these videos were between 2 to 22 seconds. The videos were shuffled, so the BUTs were not ordered in any particular way. The clinicians were asked to view each video once with a stopwatch and write down the moment the break happens (or equally could press the stop button in the video controls on the computer). This request should imitate the way the clinicians measure the BUT in the clinic. This value is denoted as the **BUT**. Then they were allowed to go over the video as many times as required and write down the range of frames where they think the break has happened and the region of the break. These values can be different from their initial value. The range of frames should be as small as possible and must include the time when they are sure the break has happened. The middle value of this range is considered to be the **Corrected BUT**. The aim of letting the clinician review his initial estimate has a two-fold purpose: (a) Learning the acceptable range of frames that are considered correct for the DEBUT. (b) In the clinic, the clinicians usually get only one chance when measuring the BUT. We would like to quantify the error of the corrected BUT compared to the original BUT, when the clinician has the option to review the video.

Four experienced clinicians, who regularly perform the FBUT test, were given the videos and we analyzed their results in Fig. 7.1. The graphs are sorted according to the root mean square (RMS) BUT, which is computed using the measurements of all four clinicians. We denote the BUT value of clinician c for video v by BUT_c^v. The RMS

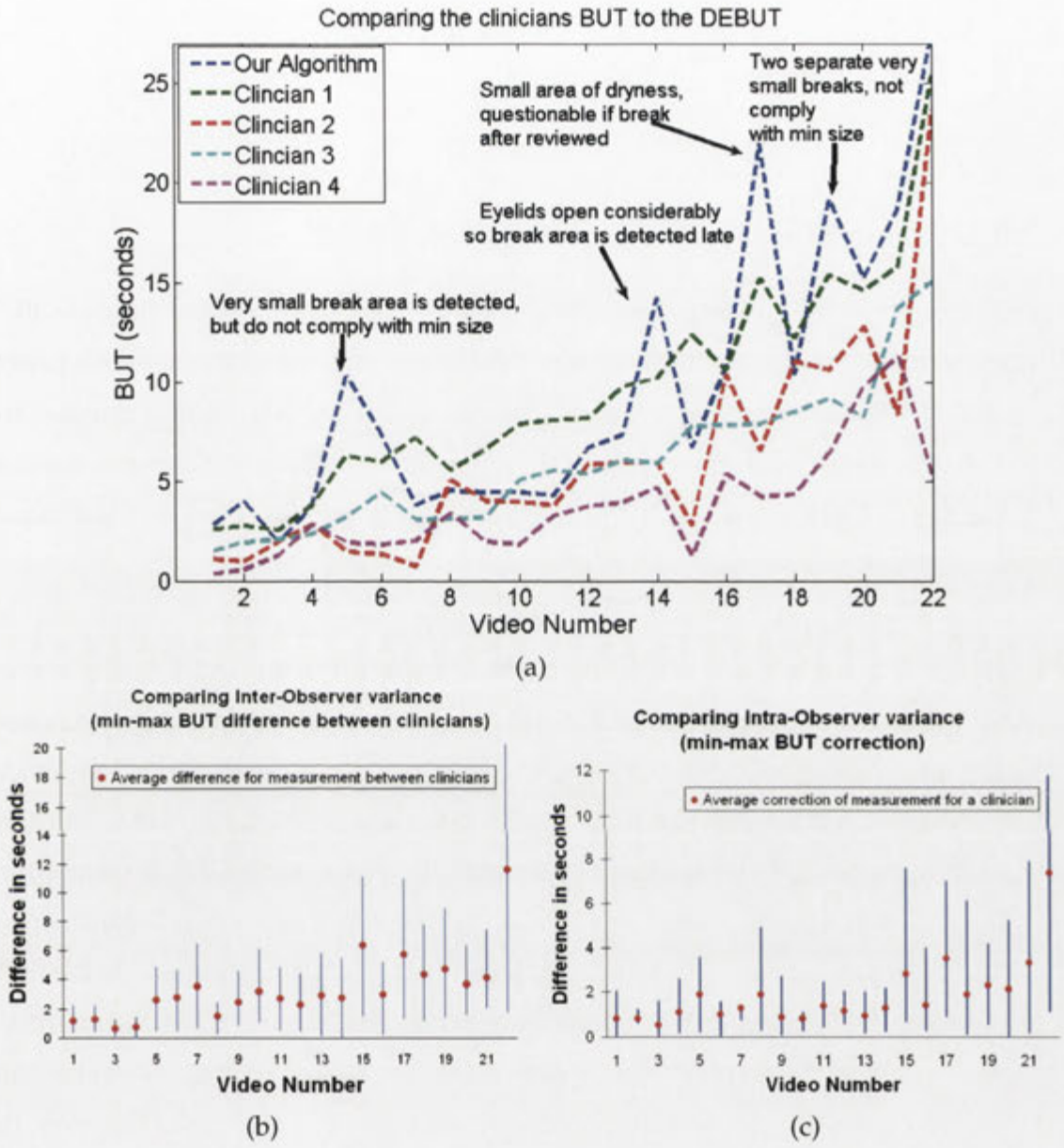


Figure 7.1: *Summary of experiment. The x-axis in all graphs is the video number. The videos were ordered according to the root mean square (RMS) BUT computed from the measurements of all 4 clinicians (a) Graph of the BUT value computed by each clinician and the algorithm DEBUT. The mere purpose of the graph is to provide indication about the large variance between the observations of the clinicians. This can be observed by noticing that no two lines follow the same pattern. The arrows point to videos of disagreement between our DEBUT and the clinicians. The reason is explained next to the arrows. (b) Inter-Observer results showing the difference in the Corrected BUT between the clinicians. Each column depicts the average, min and max value of the absolute difference between any two clinicians. (c) Intra-Observer results showing the average, min and max value of the absolute difference in the estimation of the BUT between the clinician's first and second measurement.*

value of video v is computed as follows:

$$RMS_v = \sqrt{1/4 \sum_{c=1}^{4} [BUT_c^v]^2}. \tag{7.1}$$

7.1.1 Comparing the clinicians BUT to the DEBUT

The graph in Fig. 7.1(a) compares the algorithm **DEBUT value** to the **clinicians BUT**. It is noticeable how the four graphs of the clinicians do not follow a specific pattern of BUT's values. We have achieved a strong correlation with the first clinician and a slightly less good correlation with the other clinicians (it varied between the clinicians). However, on average it is not worse than the correlation between the clinicians as can be seen in part (b).

In a few videos, our DEBUT was on the upper range given by the clinicians. We speculate that it is a result of our requirement that there is no further major decrease in the intensity of the pixel (requirement 2 in subsection 5.5.2). In addition, two videos had only a very small area of break forming in the superior part. Our algorithm did not detect a full thickness break in these videos, since the requirement in (5.12) regarding the minimum break size did not hold. In these cases, a DEBUT value which equals to the sequence's length is assigned. An example of such video is shown in Fig. 7.2. The figure contains 5 images from video number 17 that correspond to the BUTs given by the clinicians and the middle image is the last image of the sequence, which corresponds to the DEBUT. Next to each image, the BUT is written. All 4 clinicians treated the small break in the superior part as the cause of the break. However, the variation in their estimations was up to 12 seconds. Furthermore, the area of break is not even that dark (compare for example with Fig. 7.5), which is why our algorithm did not discover it as a break.

In one of the 22 videos, there was noticeable fluorescein spreading after the blink (thus after starting to record the video). This caused the brighter fluorescein to move upwards, making some areas darker just because of the movement and not due to dryness. As a result, these areas were eventually detected by our algorithm as break areas. However, the DEBUT value was still in the range of BUTs values given by the

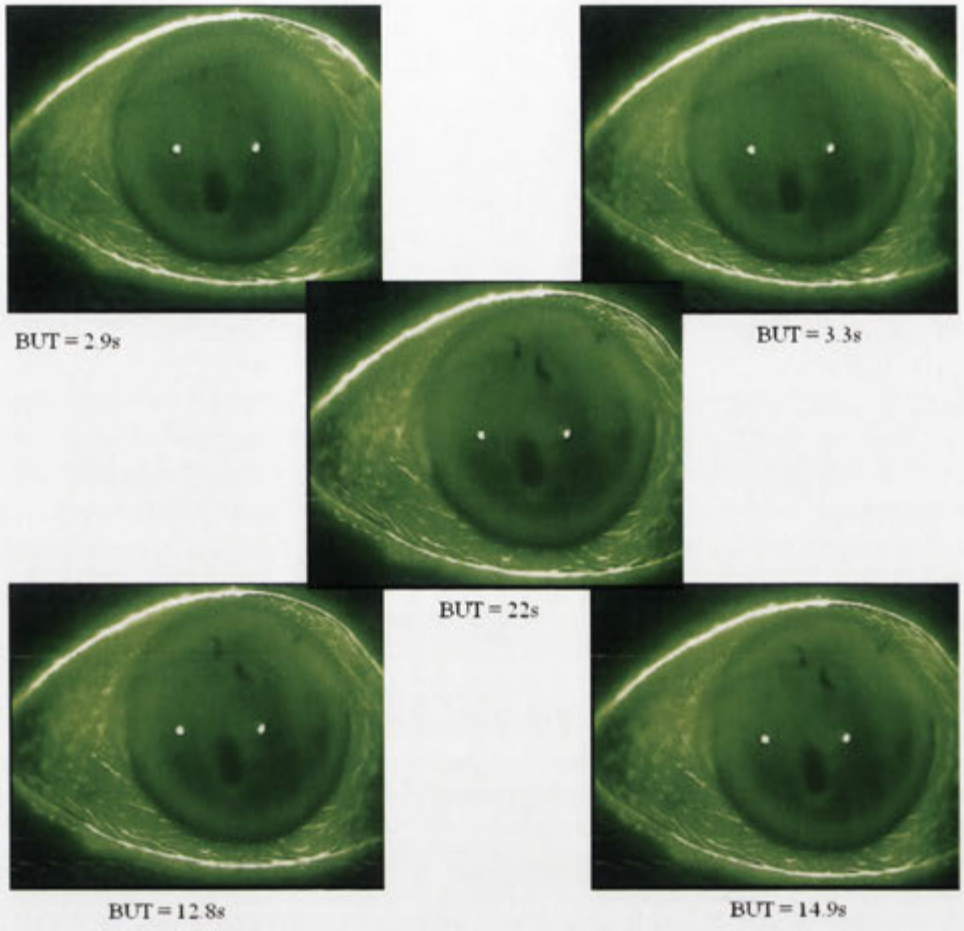


Figure 7.2: *Images corresponding to the BUT given by the clinicians and the software (in the middle) to video number 17. The video has an RMS BUT of 10.5 seconds.*

four clinicians. This is demonstrated in Fig. 7.3 by using the aligned video, so the iris is fixated. The circle highlights the area that became exposed when the fluorescein has completed its spreading at time 3.2 seconds. As a result at time of 8 seconds, this area was considered as a break region. We noted that in most cases, bad fluorescein spreading is not a problem, and can usually be avoided by asking the patient to blink a couple more times before recording the video.

An interesting observation is that the BUT value given by the clinicians in some cases was related to the time when dark areas just started forming and not necessarily when they were the darkest. Without judging whether this is the correct way to estimate the BUT or not, it creates large discrepancies between the clinicians. This raises

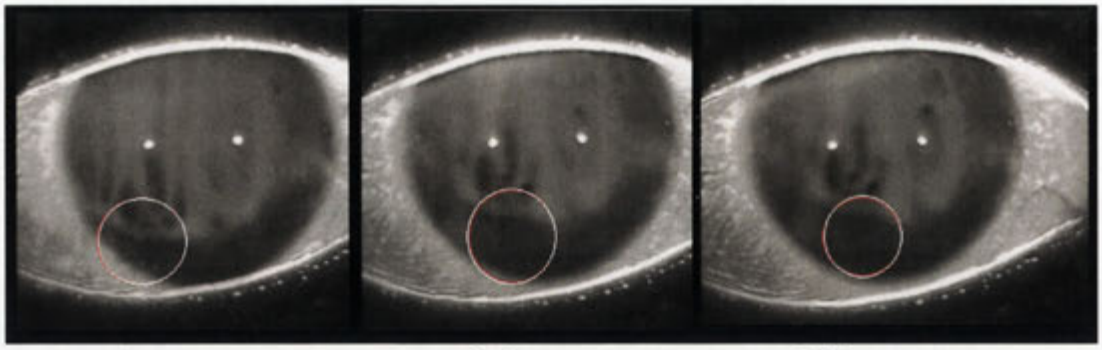


Figure 7.3: *A video with bad fluorescein spreading. The images are taken from the aligned video, so the iris is located at the same location. The first image is the image immediately after the blink. The second image is at the time the fluorescein stops spreading. The last image is the image that corresponds to the DEBUT. It can be observed how the fluorescein moves upwards between parts (a) and (b) and creates artificial dry areas.*

a couple of questions: first, how the clinical definition of the BUT can be improved to create a more unified estimation? Second, how blackness is perceived, especially if the original image is already quite dark. In such cases, regions on the cornea can appear darker, even when only a small decrease in intensity has actually happened.

We also examined the region of the break (see CCLRU standards [105]) that was provided by the clinicians as the region where they first detected the break. In sequences that had only one major break area, the clinicians always agreed between themselves. There was a variation in the location of the break when a few breaks progressed simultaneously. In some cases it could be correlated to the difference in the estimated BUTs. Fig. 7.4 demonstrates an interesting case where the four clinicians pointed towards three different areas. However, the difference in the BUT value between the first and second clinicians is only 0.7 seconds (11.6s vs 10.9s), but their detected breaks were in different quadrants as highlighted by the circle.

Comparing the DEBUT when using the monotonic constraint

In chapter 6, we presented an extension to the dryness algorithm from chapter 5. The video was transformed into a 3D volume and we incorporated a monotonic temporal constraint on the pixels' intensity in the dryness image. We demonstrated that the

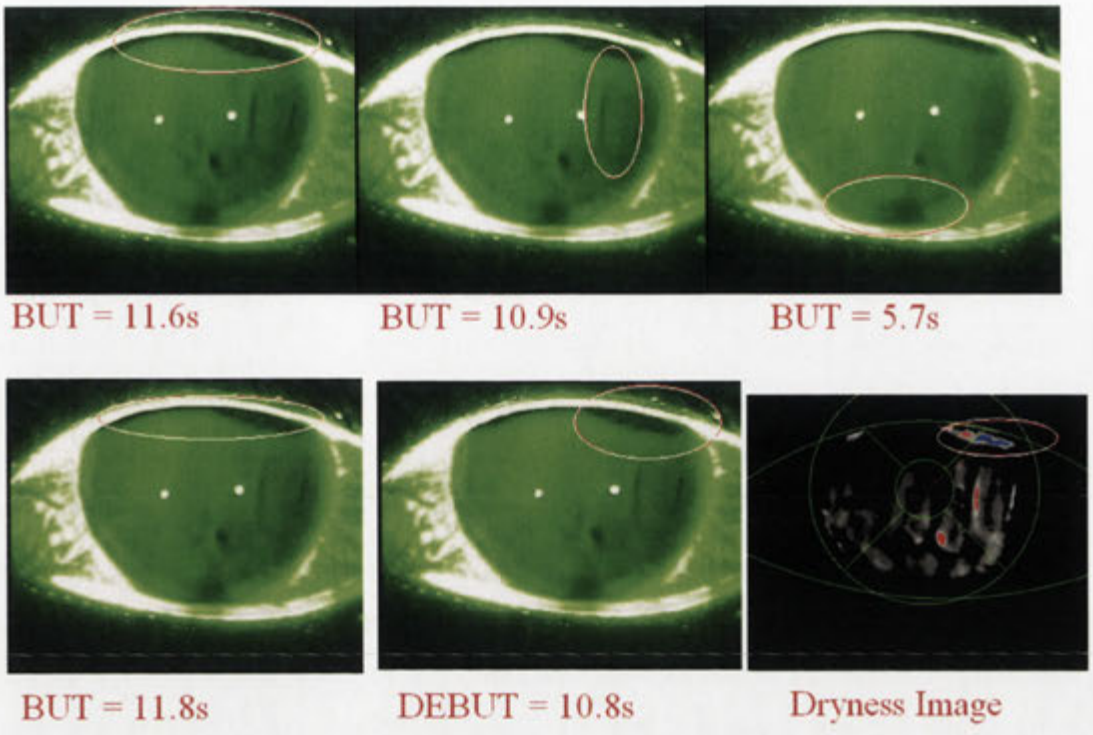


Figure 7.4: *Demonstrating the areas that were given by the clinicians as the area where they first detected the break. Although some of the BUTs are quite close to each other (first and second images), the clinicians detected a break in a different region. The detected areas were the following: superior, temporal, inferior, superior and superior (by the software). The dryness image produced for this video is in the lower right image and the region that caused the break is colored in blue.*

results were less sensitive to noise and resulted in a smaller number of label changes. In this section, we are interested in validating that the DEBUT value, when computed using the monotonic constraint, is still valid.

To that end, we used the same set of 22 videos, which all of them have a break. When using the monotonic algorithm, a pixel is considered as having a break if it is assigned the highest label in $\mathcal{L}$. Following the discussion in section 6.6, we have also decreased the threshold A_{DEBUT} that defines the minimum size of a break from 0.0025 to 0.0015. By doing so, we hope that small breaks that were not discovered using the regular method will be detected using the monotonic algorithm. However, we also ran the risk that breaks will be discovered earlier, thus a lower DEBUT value which might differ from the clinician's BUT.

We tested the monotonic algorithm using the 22 videos and recorded the DEBUT.

We first compared the monotonic algorithm with the original one. The absolute difference in the DEBUT value between the two methods was 0.76 ± 0.88. Therefore, the differences can be seen as negligible and no major change in the DEBUT value exists. We note that the DEBUT value can be adjusted to closely match the DEBUT value of the original algorithm (if needed). For example, changing the way dryness intensities are associated with labels in (6.13) or adjusting the thresholds used to compute the dryness intensity $\tilde{I}_t(x, y, t)$ in (5.8). However, we explicitly did not change these thresholds, so the results are achieved with the same parameters and also due to the uncertainty what is the gold standard because of the large variation in the BUT values.

When we compared the results, we found that the main advantage of the monotonic algorithm lies in the lower break size. The monotonic method was able to correctly detect small breaks in two videos which were not discovered using the original method. Images from one of these videos are depicted in Fig. 7.6, where one can see how small the break areas are. In both videos, the computed DEBUT was approximately 2 seconds earlier then the original method (where the DEBUT was the video length). Thus, the contribution is double fold: (1) The DEBUT has a better match with the BUT provided by the clinicians (2) A break was discovered which provides clinically important information. The detection of these breaks is directly attributed to the lower minimum break size. If we used the lower break size with the original method, it would have detected these breaks as well. However, since the original algorithm is more sensitive to outliers, it incorrectly computed an early break in two other videos. Therefore, the monotonic algorithm manages to correctly compute the DEBUT and also detect smaller breaks.

7.1.2 Evaluating inter-observer variance

The inter-observer graph in Fig. 7.1(b) compares the **absolute difference** of the **Corrected BUT between clinicians**. As the experiment had 4 clinicians, we computed the difference for the 6 possible ways of comparing between two clinicians: $\binom{2}{4} = 6$. Each column in the graph shows the average, minimum and maximum of this difference. For example, let us assume that for a certain video, the BUT values given by the clini-

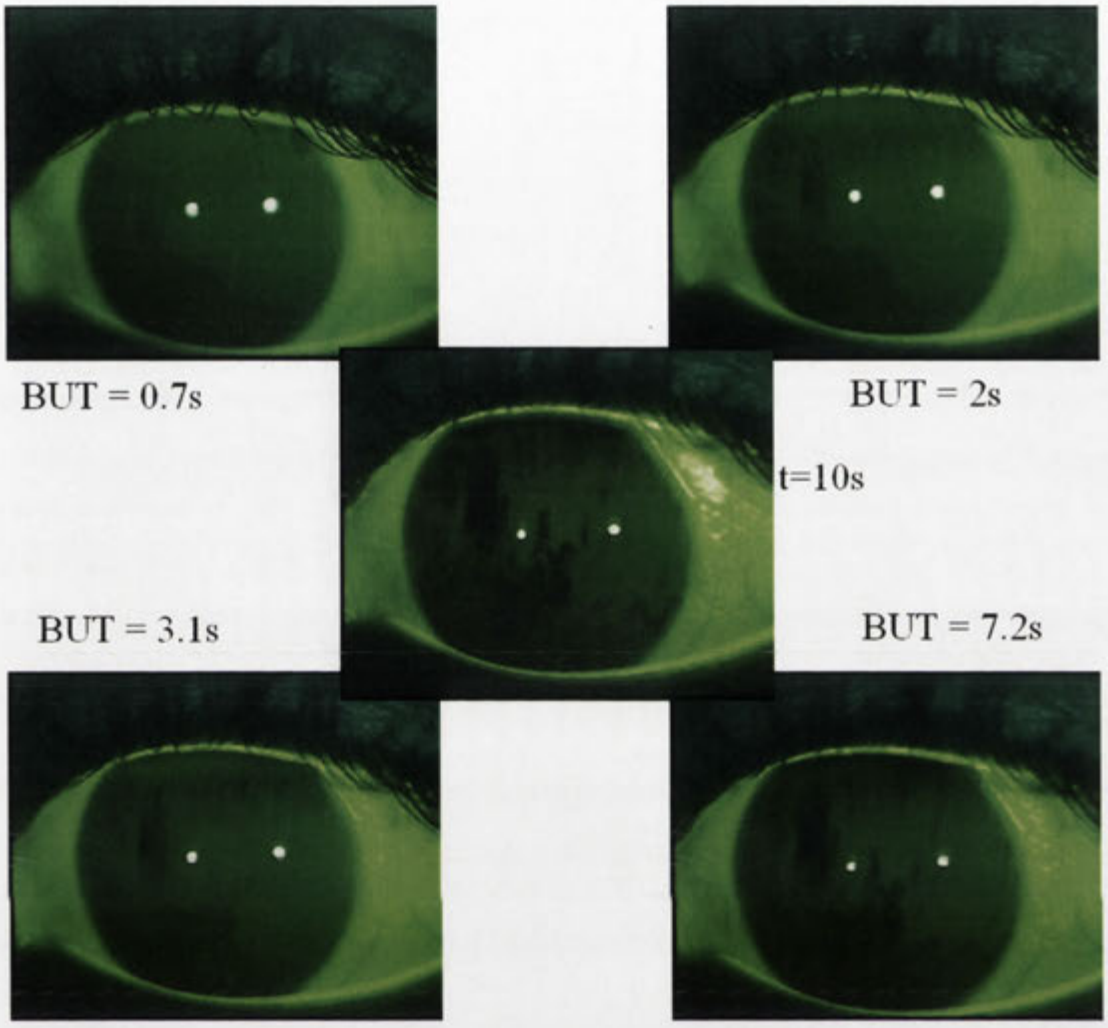


Figure 7.5: *Images corresponding to the* ***Corrected BUT*** *given by the four clinicians to video number 7. The middle image is the image after 10 seconds showing that breaks started forming in other regions as well, and possibly regions where the early break formed are still becoming darker. The video has a relatively short RMS BUT of 4.5s seconds, but the estimates vary from 0.7s-7.2s.*

cians are: 1s, 4s, 5s and 7s. Then the minimum difference is 1s (5-4), the maximum difference is 6s (7-1) and the average difference is: $1/6 * (3+4+6+1+2+3) = 19/6$s. The graph shows that meaningful differences exist between the clinicians. We also compared the standard deviation of the BUT value among the clinicians for the 22 videos. We found out that the standard deviation is higher than two seconds in 15 videos and higher than 3 seconds in 6 videos.

Examining the maximum difference between the clinicians reveals that it can be

over 10 seconds. The largest difference occurs on video 22 (see Fig.7.6). Even if we consider the small BUT provided by clinician 4 as an outlier, there is still a large difference in the BUT estimation between the first and the second clinician. However, the big differences in BUT are not necessarily for videos with a high BUT value. For example, video number 7 with RMS BUT of 4.5 seconds had an extreme difference of 6.5 seconds between two clinicians and a standard deviation of 2.8 seconds. Even this small sample demonstrates how subjective the BUT value is. Fig. 7.5 shows images from video number 7, which correspond to the clinicians Corrected BUT. Interestingly all clinicians reported the break in region 2 (nasal), which is in the left hand side in these images (as it is a left eye). Looking at the 5 images, it is clear that the size and degree of blackness of the break is quite different in each image. This demonstrates again the difficulty of deciding what is the exact moment an area is becoming dark enough.

7.1.3 Evaluating intra-observer variance

The intra-observer graph in Fig. 7.1(c) compares the **clinician** first measurement (**BUT**) with his corrected one (**Corrected BUT**). For each video and clinician, the absolute difference between the clinician's BUT and Corrected BUT was computed in seconds. Each column shows the average, min and max value of this difference for that video (among all clinicians). The results were quite surprising, generally showing a high consistency between the two measurements of the same clinician. The first two clinicians have corrected their results in a value of less than 1s for about half of the videos. This shows that they had a high confidence in their estimation in these videos. The location of the red dot in each column of the graph corresponds to the average correction among the four clinicians for that video. It can be seen that on average, the change is not big in most videos.

A very interesting result is that the max correction for a clinician was between 7 to 11 seconds. For all clinicians, it was at the same video - number 22 (last column in the graphs). Correlating it to the results in parts (a) & (b), we find out that correcting the BUT value did not make the BUT for this video more unified among the clinicians.

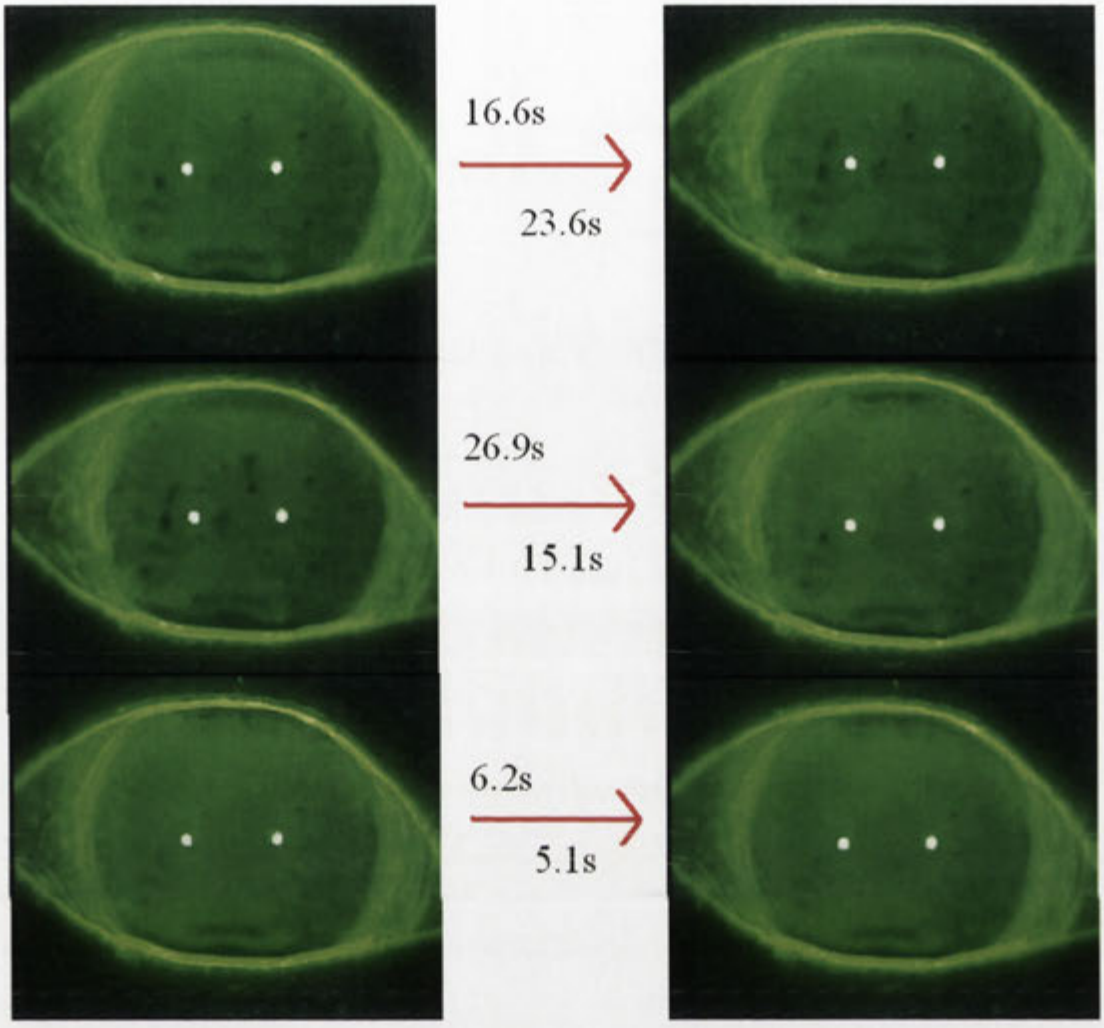


Figure 7.6: *Demonstrating the extreme case of intra-observer correction. The images are from video number 22. The images on the left correspond to the clinician's BUT value and the images on the right correspond to the clinician's Corrected BUT. Two clinicians increased their BUT value drastically (see first row), while the third decreased it drastically (see second row). The fourth clinician did not change its BUT that much (see third row), but its BUT value is very different from the other clinicians.*

This is demonstrated in Fig. 7.6, where two images are shown for each clinician: the clinician's BUT image and the Corrected BUT image. Clearly, the break is related to the small circular dots appearing in the temporal and nasal parts. However, the degree of blackness of these regions only changes very gradually over a long period of time. This finding shows that the clinicians face a real difficulty in assessing the BUT at specific videos such as this one.

Another phenomenon is that in some videos a break was developing in the superior region, close to the eyelids, and the clinicians usually corrected down their BUT value in these videos. This is probably because it is easier to focus only on the central area at the first measurement. When given the opportunity to review, the iris can be observed more thoroughly for the superior part as well.

7.2 Comparing the DEBUT and location of break to a single clinician

We compare the BUT value and the reason of the break provided by a very experienced clinician to our computed DEBUT. The aim is to show that the results are in agreement when compared to a single clinician. We asked the clinician to analyze a set of 21 *new videos* and write down the BUT, reason of the break and the area of the break. All videos were different from those that we used in the previous section. The possible reasons for the origin of the break were: no break, a corneal break and a break originated near the upper or lower reservoirs (black line). The chosen videos had a varying BUT, where about one third of the videos had no full thickness break . This depicts the everyday case, where not all patients have a corneal break. About one third of the patients had some dryness symptoms related to the black line. When the clinician or the algorithm reported no break, we considered the DEBUT to be the sequence length (for comparison reasons). The average length of the videos is 12 seconds ranging from 6 to 20 seconds, and the average BUT reported by the clinician was 10 seconds.

Fig. 7.7 shows a graph of the clinician's BUT and the algorithm detected DEBUT on all 21 videos. The videos were sorted according to the BUT for presentation purposes. The average difference between the measurements is $0.7s \pm 1.1$ seconds and is well inside the acceptable range taking into account that clinicians measure the BUT in seconds. Figures 7.8(a) and (b) show the images corresponding to the BUT and the DEBUT for the first video respectively. This is the video with the largest difference between the clinician and the algorithm. As can be seen in the figure, the area of the break is very small, which is why it was not detected by our algorithm in that early

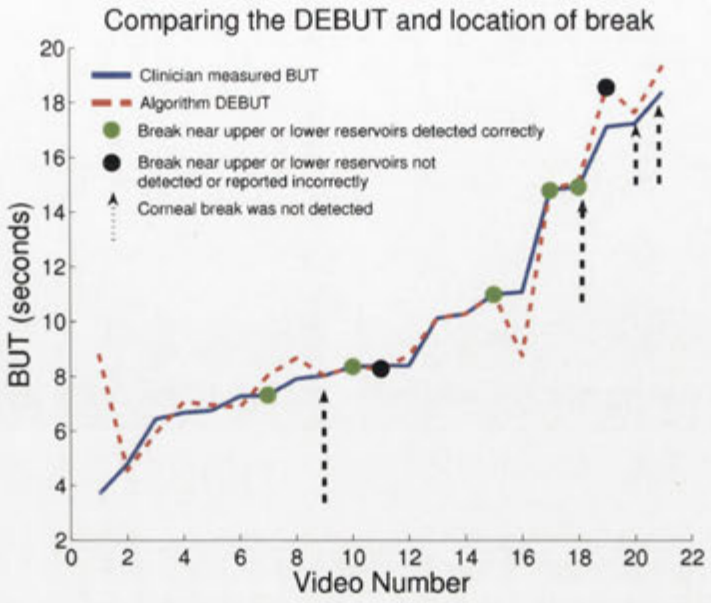


Figure 7.7: *Graph comparing the clinician BUT to the algorithm DEBUT in 21 sequences. Green circles indicate videos where both the clinician and our algorithm indicated a reservoir related break, while black circles indicate videos of disagreement. The black arrows point towards videos that our algorithm reported no corneal break, but the clinician did. One can see that in these cases, the error is small, showing that the break happened late in the sequence.*

stage.

In 4 videos, the clinician indicated an upper reservoir related break and in 2 other videos indicated towards both the upper and lower reservoirs. Five cases were detected correctly as depicted in the graph by green circles. The black circles show the two cases of disagreement: in the first, an upper reservoir break was not detected by the algorithm and in the second, our algorithm reported an upper reservoir break. In three cases, the clinician pointed out the reservoir related break as the only reason of the break and so did our algorithm (therefore, DEBUT equals to clinician BUT). Corneal breaks were not detected for 4 patients (and we assigned DEBUT = sequence length), depicted by the black dashed lines. However as seen in the graph, the differences ($DEBUT - BUT$) are small ($0.6 \pm 0.3s$). This difference means that the break occurred late in the sequence and was probably not detected due to the area not becoming dark enough or being too small. As at that stage, the patient blinked, we cannot say if the error would have been increased or not. However, in the given videos,

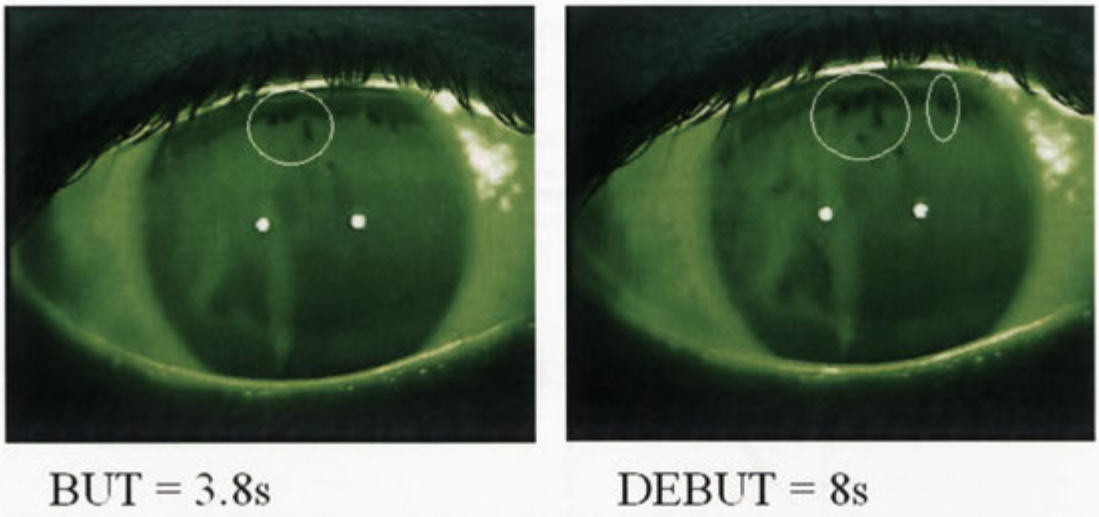


Figure 7.8: *Images depicting the largest difference between the clinician's BUT and the algorithm's DEBUT. The images correspond to video number 1 in the graph of Fig. 7.7. The reason for the difference is the small size of the break area. The white circles enclose the area where the break was detected by the clinician (left image) and by the algorithm at the time of the DEBUT.*

we can conclude that the incorrect detection is less significant.

7.3 Repeatability experiments

We have carried out an experiment to try and prove the repeatability of our method in terms of DEBUT and area of the break. We chose 5 patients for this experiment who were willing to be examined in two separate days. We followed the clinical routine described in section 2.3.4 and captured 1 or 2 sequences for each eye. A few hours later, we repeated the same test and then twice again on a different day. For each patient, we captured up to $2 \times 2 \times 4 = 16$ sequences and in total 64 sequences. We used 59 of them for our analysis and as for the remaining 5, the patients did not open their eyes enough or the blink-to-blink intervals were very short. We were interested in the BUT and the area of break for each patient at the 4 different times. Our goal is to demonstrate that the symptoms and their analysis are repeatable for the same patients. For example, we would like to show that the BUT happens approximately at the same time with a similar shape at different sequences of the same patient.

We analyze and present the results of the experiment in Table 7.1. The results

reveal that 2 patients and in total 21 sequences had no dryness symptoms at all. The third patient had only 1 sequence out of 14 in which he had a full thickness break and a detected DEBUT of 5.4s. In the next sequence immediately following this one, he developed some reflex tearing that might have inhibited more breaks. This patient always had large blink to blink intervals of at least 12 seconds, except for the break sequence which had a break after 8 seconds.

The fourth patient had more varied results. Nine of the sequences had no break, although thinning of the tear film was evident. In 3 sequences, breaks existed and were mostly very small in area and appeared in different locations. The clinician noted that these breaks were not very significant. Two sequences had symptoms related to the upper and lower reservoirs, i.e. black line symptoms.

The fifth patient had dryness symptoms related to the upper and lower tear reservoirs in 2 sequences (thinning areas in the inferior and superior parts), and one case of a corneal break. In the other 7 sequences, the clinician did not indicate any breaks, but just thinning of the tear film. The reservoir related breaks occurred only twice, so it is impossible to derive any conclusions.

Since our chosen subjects mostly did not show dryness symptoms, more clinical experiments are required to draw conclusions regarding corneal breaks. In a controlled clinical research, the patients should be chosen according to existing dry eye symptoms.

Some conclusions can still be drawn from the experiment. The first three patients did not show any breaks at any test (except for one measurement), so we notice the repeatability of that result. Reservoir related breaks were partially repeatable. When patients 4 and 5 had breaks, they were mostly horizontally shaped breaks in the inferior and superior parts. Even though they were not associated with the reservoir by the clinician, they occur in similar regions and resemble the pattern of the black line. Finally, the blink to blink interval was quite repeatable as well, except for the first patient.

Table 7.1: *Summary of the repeatability test with 5 patients and 59 videos. 'None' is the number of sequences with no breaks at all, Reservoir Break is a break originated near the the tear reservoirs and Corneal Break is a break at any region on the cornea (excluding reservoir breaks). The Avg blink-to-blink is the average blink to blink interval in seconds. The table shows the analysis provided by the clinician. We also analyzed the break patterns for patients no. 4 and 5: Patient no. 4: 1 Superior (Horizontal), 2 Inferior (Pool), Central (Pool), Nasal (Pool). Patient no. 5 had one corneal break in the inferior region. Large blink to blink intervals are usually related to non-dryness symptoms. It is also of interest to measure the break-to-blink interval, as this is the period where the patient feels most inconvenient. However, it was relevant only in 4 cases in this experiment.*

No.	None	Corneal Break	Reservoir Break	Avg blink-to-blink (s)
1	12	0	0	11.5 ± 8.7
2	9	0	0	27.6 ± 3.9
3	13	1	0	22.1 ± 8.8
4	9	3	2	17.0 ± 10.7
5	7	1	2	10.3 ± 2.3

7.4 Conclusions

In this chapter, we have compared some of the dryness symptoms that our algorithm detects to the clinicians' through a set of videos with and without a break. We compared the DEBUT value of the algorithm to the clinicians' BUT value and the region of the break. We showed that the DEBUT value is in the acceptable range provided by the clinicians and also reservoir related breaks are detected. As presented in chapter 5, our algorithm provides more data, such as the degree of thinning, the area of the break, the shape and the progress of the break and is not based only on a clear-cut definition of dryness such as the BUT value. However, it is hard to compare most of this information to manual results.

We believe that another big contribution of this chapter is the demonstration how subjective and hard it is to estimate the BUT. We showed that even when using only four qualified clinicians and 22 videos, large discrepancies exist between the measurements of the clinicians. It seems that each clinician tends to estimate the FBUT according to his own experience. Therefore, it raises the question, what measurement should be considered as a ground-truth when we compare to our results.

We could also learn a lot from the Corrected BUT values provided by the clinicians

after they were given the opportunity to review the video. Although in most videos the clinicians did not change their BUT significantly, in some cases the revised value was quite different from the original one. We speculate that this is related to the slow progress of the break or to the location of the break not in the central part of the cornea. Both assumptions are based on analyzing the BUT value and area that were given by the clinicians and observing the current clinical conventions.

The results obtained in the repeatability tests were inconclusive as most patients had no dryness symptoms. Further tests are needed using a sample of patients who have dryness symptoms. This can be sufficient to try to correlate the DEBUT to the region, shape and progress of the break. This is assuming that the breaks actually repeat themselves in sequences obtained at different times.

Some of the differences between the clinician BUT and the software DEBUT could be bridged by making the algorithm more sensitive to the minimum break size defined in (5.12). As a result, break areas that stay small or increase slowly (as in Fig. 7.8) will be detected earlier. We showed that the extended monotonic algorithm handled such break patterns better. The lower minimum break size did help to match the DEBUT value with the clinicians BUT value. In addition, the increased sensitivity to outliers did not produce wrong DEBUT values when compared with the original method.

Finally, all these results probably reinforce what is already known - the FBUT test is not reliable by itself as a measurement for dryness, but is still the most commonly used test by the clinicians. Our contribution is offering a new measurement, the DEBUT, which is robust and not operator-dependent. With a little bit of tuning, it can be very valuable to the clinician.

Chapter 8

Detection of the Tear Meniscus Shape

In this chapter, we present an algorithm to evaluate the tear meniscus quality. Assessing the tear meniscus in conjunction with the detection of dry regions and the DEBUT provides the clinician a detailed clinical diagnosis regarding the degree of dryness and the cause. See section 2.3.5 for more details.

Our algorithm segments the tear meniscus and analyzes its irregularity. We use the same videos that are recorded for the analysis of the break in chapter 5, but analyze a single image taken two seconds after the blink. This timing is chosen as the meniscus shape tends to stabilize two seconds after the blink, and provides the most accurate information.

As mentioned in chapter 2, the recent report of the International Dry Eye Workshop (DEWS) [18] mentioned the lack of a gold standard for diagnosis of dry eye and the usefulness of performing more than one diagnostic test. In this chapter, we analyze the tear meniscus shape in our fluorescein images and provide its analysis in conjunction with the tear film break pattern of chapter 5.

The segmentation result of a regular meniscus will be a connected object with no gaps, (see Fig. 8.1(b)). However, this cannot be enforced easily when using graph-cuts (see section 4.5) due to the locality of the pairwise term. To overcome that, we use an energy function with asymmetric pairwise terms to enforce continuity. We incorporate a directional constraint which distinguishes between upwards and downwards moves away from the center of the segmented region. Only one change from object

to background is permitted when moving up or down from the center. Thus, pairwise terms in the cost function are not only asymmetric but also different for points above and below the eyelid. The eyelid curve (known as a shape prior) is modelled using a spline. Our approach simultaneously optimizes the eyelid curve, which defines the asymmetry, and segments the meniscus. Therefore, segmenting a long (thin) connected object is actually encouraged by the shape prior. The best shape prior and the segmentation are produced by iterative optimization over the parameters of the eyelid curve. Within each iteration, the energy function value is determined using graph-cuts.

8.1 Introduction

The tear menisci (or tear reservoirs) are situated along the upper and lower eyelids and are part of the exposed tear volume (see Fig. 8.1(a)). The height of the tear meniscus is an important measure of the volume of fluid in the tear film. A shallow or irregular meniscus is highly significant because it often results from the many risk factors associated with dry eyes [84]. Lid-parallel folds (or conjunctival folds or branching) [30], see at Fig. 8.1(c) that the lower meniscus has another line originating from it in the center of the image, are common with age and are also associated with dry eyes. These folds are hard to detect as they can be hidden by the meniscus. Such folds will deprive the tear film of the important fluid and in severe cases can come into contact with the cornea. The detection of these irregularities is especially important before fitting contact lenses. Assessing the quality of the meniscus is usually not carried out by the clinician as it is a hard task, especially over the conjunctiva. More information about the meniscus can be found in section 2.3.5.

8.1.1 Previous work

A new approach to tear meniscus segmentation uses Optical Coherence Tomography (OCT) (see chapter 2.3.5) to assess the tear meniscus height and curvature [109]. An OCT slice provides information regarding the area of the tear meniscus at that location only, but the images are sagittal cross-sections through the eye – quite different

(a) (b) (c)

Figure 8.1: *(a) An image taken from the video two seconds after the blink. Notice the uneven illumination between the left and right sides of the lower meniscus. The upper meniscus suffers from bad illumination in both sides. (b) Segmentation result of the menisci. The lower meniscus is mostly continuous with varying heights becoming thinner on the left hand side further from the iris. (c) An example of an irregular lower tear meniscus containing folds.*

from our frontal images. In order to estimate the full shape of the meniscus, many slices have to be taken. OCT imagery is mainly used for viewing the fundus and conjunctival folds cannot be detected. The routine use of OCT by clinicians to detect DES is uncommon, especially when compared to the prevalence of a slit-lamp diagnosis.

Using graph-cuts to segment medical images is common (see chapter 4.5). Recent research focused on joining shape information with energy based cost functions. Kumar et al. [70] address the problem of incorporating a shape prior and note that energy functions based only on MRFs do not usually give rise to realistic shapes. In [61] the authors improve this work by using dynamic graph cuts. Their aim is to segment the human body from a given image using graph-cuts and also estimate its pose. They introduce a stick-man model, which is a thinned image of the human pose defined by a set of parameters. The shape prior is defined as a distance transform to the stick-man and is incorporated as a unary term. As a result, segmentations that follow the pose of the stick-man are favored. The segmentation process iterates with different model parameters until reaching a minimum, which provides both a segmentation and the closest human pose (to the given image).

In [3] the authors segment the kidney using graph-cuts and introduce a shape prior by building a model of iso-contours around the kidney in training images. The shape constraint is incorporated as a function of the distance of the pixel from this model.

In a similar way to [61], the constraint is defined as a unary term. Such solutions are useful to couple the distance of a pixel from the model with a MRF (see Section 8.3.3). However, they are insufficient to enforce a sense of direction in the segmentation process.

Few segmentation applications combine a shape prior as part of the pairwise term. Boykov et al. [15] segment the liver from the muscle by using an asymmetric pairwise function. Their function is dependent on the pixels' intensities and encourages a cut from bright object to darker object. Nevertheless, it does not incorporate information about the desired shape of the object.

We note that early graph-cuts literature mainly used the Potts model or a similar cost which is not suitable for segmenting elongated thin objects (such as the tear meniscus). This shortcoming is solved by our use of a long thin shape prior. This ensures there is no truncation of long shapes. Thus, the method can be thought as an enhanced active shape model method [25]. However, our implementation adds another dimension to the problem: asymmetry and the use of a shape prior. Our cost enforces only one move from object to background and vice versa, so segmenting a long thin object is actually encouraged. Kohli et al [60] also segment thin long objects (such as the legs of a cow) using high order potentials and graph-cuts.

Graph-cut solutions using asymmetric cost functions have been discussed in chapter 6. As we mentioned there, asymmetric costs have not seen widespread use. Some recent publications use directed edges to enforce directional constraints [15, 59, 79, 121] or apply asymmetric penalties which depend on the potentials associated with the random variables [99].

In this chapter, we show how to combine a shape prior with a directional segmentation into a graph-cuts based solution. We build an asymmetric pairwise cost function, which determines the direction of the segmentation according to the shape prior. We first describe how the model is estimated from the given image and then elaborate how the pairwise and unary terms are built. Finally, we present the segmentation results and analyze them.

8.2 Estimating the shape prior

This section shows how an initial estimation of the meniscus shape prior is found. We notice that the accurate detection of the eyelids provide a good initial estimation for the meniscus. Given an image $I(x, y)$ of size $X \times Y$ (such as in Fig. 8.1(a)), we use a very similar technique to the one in chapter 5.3 to fit curves to the upper and lower eyelids. The limitation of the method from chapter 5.3 is that the fitted curves are usually very accurate over the iris area, but not always over the conjunctival area. In areas further from the iris, the eyelids can curve quite strongly, thus fitting a single quadratic polynomial is impossible.

The only difference in the fitting technique we employ in this chapter is that we fit more than one polynomial to the eyelids. We divide the eyelid curve to n_s non-overlapping (or with a small overlap) segments over the image horizontal domain $[0, X-1]$. Then we use RANSAC to fit a quadratic function to each segment resulting in the following function:

$$R(x) = \begin{cases} a_1x^2 + b_1x + c_1 & \text{if } 0 < x < X/n_s \\ \vdots & \vdots \\ a_nx^2 + b_nx + c_n & \text{if } ((n_s-1)/n_s)X \leq x \leq X-1 \end{cases} \tag{8.1}$$

Here, R is a one-dimensional function of x, where x is the location in the horizontal axis. In this chapter, we have used $n_s = 2$, and find equations for both the lower and upper meniscus curves.

In order to produce a smoothed curve that can be modelled easily, we perform a cubic B-spline interpolation [94]. We define $m+1$ uniformly spaced knots over the domain $[0, X-1]$, such that $t_0 = 0$, $t_m = X-1$ and $t_i = iX/m$ for $1 \leq i \leq m-1$. The equivalent $m+1$ data points of the eyelid curve at these knots are:

$$(0, R(0)), (t_1, R(t_1)) \ldots (t_m, R(t_m)).$$

Here, we have used the location of the eyelids curve at $m+1$ points as was computed in (8.1). We interpolate the B-spline curve $M(t)$, such that $M(t_k) = R(t_k)$ for $0 \leq$

$k \leq m$. Since the degree d of the spline is 3, we follow the convention and define d additional knots at t_0 and t_n, so $M(t)$ can be computed at the first and last intervals. The problem can be formulated as to find the coefficients c_j to $m+d-1$ cubic B-spline basis functions $(N_j^d(x))$, such that:

$$M(x) = \sum_{j=0}^{m+d-1} c_j N_j^d(x). \tag{8.2}$$

The solution is found easily using matrix computation. This whole process is the standard B-spline interpolation. The meniscus curve is now parameterized and its parametrization is denoted by θ_m. We note that the B-spline does not have to be accurate at all knots (and usually will not be). A better estimation of the prior can be found in the iterative process described later.

8.3 Using graph-cuts

Our segmentation solution is based on graph-cuts. The problem formulation was presented in chapter 4 and we repeat it briefly here: Given an image $I(x, y)$, we define the set $\mathcal{P}$ to contain all pixels in the image. The aim is to assign a binary labelling to all pixels $p \in \mathcal{P}$ as either background or object. The vector $\mathbf{x}$ is defined as a possible labelling. The assignment of a pixel $i \in \mathcal{P}$ is notated by $\mathbf{x}_i$, where $\mathbf{x}_i = 1$ assigns the pixel as Object (belonging to the meniscus) and $\mathbf{x}_i = 0$ assigns the pixel as Background. The energy function is defined as (4.8):

$$E(\mathbf{x}) = \sum_{i \in \mathcal{X}} E_i(x_i) + \sum_{(i,j) \in \mathcal{N}} E_{ij}(x_i, x_j). \tag{8.3}$$

and can be solved by using graph cuts. The graph is created using all pixels in $\mathcal{P}$ as vertices and two additional vertices: 0 and 1. Initially, the vertices are connected to each other using the 4-connected neighborhood system $\mathcal{N}$, and also all vertices are connected to 0 and 1. The costs on the edges are determined using the unary terms $(E_i(x_i))$ and pairwise terms $(E_{ij}(x_i, x_j))$. The unary and pairwise terms are also called unary and pairwise potential functions respectively. After running the graph-cut al-

gorithm, each vertex is connected to either 0 or 1 but not to both, hence dividing the image to object and background and providing a segmentation. More details about the graph creation were described in chapter 4.

In chapter 4, we showed the correlation between MRFs and graphs. Given an image (lattice), it could be equally represented as a graph or as a MRF. We showed how to solve the labelling problem by using graph-cuts. The graph-cut algorithm minimizes the joint probability over observations (pixel values) and label sequences (a labelling). When using MRFs, it is not possible to represent joint probabilities between random variables that depend on the observations. Therefore, when using MRFs, the pairwise term in (8.3) cannot depend on the data, and thus information such as the intensities of pixels i and j cannot be used.

This limitation when using MRFs led to an alternative called Conditional Random Fields (CRFs) [71] . A conditional model defines the probability of possible labellings given an observation sequence. The conditional probability of a labelling can depend on any features of the observation sequence without assuming any prior knowledge regarding their distribution. For example, in (8.3) the probability of assigning labels x_i and x_j to pixels i and j can depend on information such as their intensities, their neighboring pixels intensities and their spatial location in the image (since we assume all observations are given in advance, i.e. the image).

In [71], the authors provide a formal definition to CRFs and show its properties. An MRF can be seen as a specific case of a CRF where the joint probability is not conditioned on the data. The segmentation solution we present in this section uses the underlying observations to define the pairwise energies. Therefore, minimizing the term in (8.3) requires using CRFs (and not MRFs). However, the graph-cut algorithm does not require changes whether the cost functions depend on the underlying model or not. We will see in this section that our pairwise term depends on a set of parameters defined by the shape prior (described in previous section). Therefore any change in parameters to the CRF induces a change in the potential functions and requires computing the cut again.

8.3.1 Incorporating a directional constraint

The main observation regarding the tear meniscus is that in the normal case, such as in Fig. 8.1(a), the shape of the meniscus is a continuous smooth object with a varying height throughout the image, which does not contain gaps. When scanned in an upwards or downwards direction, only a single transition from object to background and from background to object is expected. In the abnormal case, such as in Fig. 8.4(a), there are spikes that originate from the meniscus or in some cases parallel lines to the meniscus. Assuming our initial estimation $M(x)$ for the meniscus curve is roughly correct, all segmented pixels should be connected to the curve in the normal case. In other words, breaking the continuity of the segmentation should be penalized by high energy terms.

The difficulty is incorporating continuity and shape priors into a MRF (or CRF) based solution, since it mainly provides local pixel information. When using a 4-connected or an 8-connected neighborhood systems (and hence only quadratic terms), the pairwise term only depends on two neighboring pixels. Therefore there is no obvious way to enforce shape constraints that depend on more than two pixels. For example, assume we have four vertically neighboring pixels p,q,r and s and we would like check if for a given assignment they represent a connected object with no gap. Let us assume a given assignment: $p = 1, r = 0, r = 0, s = 1$. Here we have a gap between two objects. Since each quadratic term depends on two pixels, we can detect a change between object and background from p to r and from r to s. However, we cannot easily conclude that there is a gap, since there is no term that depends on all four pixels.

In order to solve (8.3), the four possible assignments between two neighboring pixels have to be defined: $E_{ij}(l_i, l_j)$, $l_i, l_j \in \{0, 1\}$. We remind the reader that the cost $E_{ij}(0, 1)$ is the cost of assigning pixel i as background and pixel j as object. Most graph-cut based solutions assume symmetry between the terms. However, in order to add a sense of direction to the CRF , we use an asymmetric approach, where:

$$E_{ij}(0, 1) \neq E_{ij}(1, 0) \quad \text{for } (i, j) \in N \tag{8.4}$$

(Note that $E_{ij}(0,1) = E_{ji}(1,0)$ always holds, as the cost of assigning i as background and j as object is the same as assigning j as object and i as background). By doing so, it is possible to distinguish between object to background transitions (or background to object) made in one direction and the opposite direction. For example, a transition from object to background in an upward direction can be encouraged by a low cost, while the transition from object to background in a downward direction can be forbidden. This cost will encourage segmenting in a downward direction (until some condition is met, which might change the cost), but not necessarily in an upward direction. This is a similar situation to the one depicted in the right side of Fig. 8.2.

Given the initial estimation $M(x)$ of the meniscus through its set of parameters θ_m, we can now enforce the continuity constraint and control the penalty for breaking it. An asymmetric cost function is useful, when there is an existing prior, which provides knowledge regarding the direction of the required segmentation. It can also be seen as an additional way to incorporate shape prior in the CRF. Next, we describe the construction of the pairwise function.

8.3.2 Computing the pairwise term

The intuition behind the pairwise term is to use the estimated location of the meniscus to enforce **continuity** in the direction of segmentation. We will assume two neighboring pixels $p = (p_x, p_y)$ and $q = (q_x, q_y)$. Given a pixel p, it has four neighboring pixels. Two of them lie on the same horizontal line as p (same p_y value) and the other two differ in 1 pixel vertically such that $|p_y - q_y| = 1$. Therefore, it is enough to consider only a horizontal and vertical case when assigning costs. We would like to estimate the following conditional probabilities for a pair of pixels:

$$Pr(\mathbf{x}_q|\mathbf{x}_p, \theta_m), \tag{8.5}$$

and convert them to energy costs. This is the probability of assigning a label to pixel q, knowing the label of a neighboring pixel p and the curve parameters.

We will now consider three different cases between two vertically neighboring pixels as depicted schematically from left to right in Fig. 8.2. Without loss of generality,

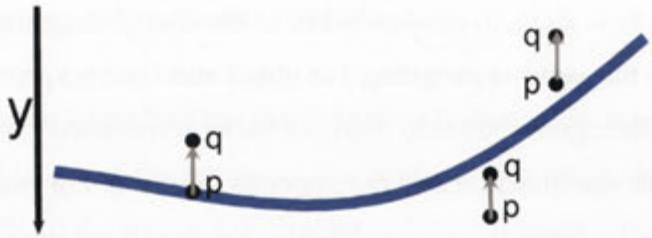


Figure 8.2: *A schematic image of the tear meniscus and three possible cases of pairwise costs. From left to right: pixel p is on the curve and pixel q is above; pixels p and q are below the curve; and pixels p and q are above the curve. We show that the associated costs are dependent on the location of the meniscus and the direction of segmentation.*

we assume that the transition is always upwards from pixel p to q (as pointed by the arrows in the figure). If the transition is downwards then the costs of changing labels are reversed. For convenience, we will omit the subscript pq from the cost terms.

Case 1: Pixel p lies on the curve, therefore our assumption is $\mathbf{x}_p = 1$. The probability of a neighboring pixel q assigned as object depends on the average height of the meniscus. This probability can be learned from manually segmented images of the meniscus: Assuming we are on the meniscus, what is the probability that a neighboring pixel is also part of the meniscus?. This probability should be relatively high as the meniscus height is usually a few pixels. The probability of assigning q as background is just $1 - Pr(\mathbf{x}_q = 1|\mathbf{x}_p, \theta_m)$. Therefore, the cost $E(1,1)$ should be low and the cost $E(1,0)$ should be high. We notice that this case is independent of θ_m.

Case 2: Pixels p and q lie below the curve. Assigning the same label for both pixels preserves the continuity of the segmentation and should incur a low cost. Assigning $\mathbf{x}_p = 0$ and $\mathbf{x}_q = 1$ does not necessarily creates a gap as q can be connected to the meniscus and therefore is expected to happen. Such a transition ends the downwards segmentation of the meniscus at pixel q at that row. This cost should be proportional to the average height of the meniscus. The interesting case is the transition $E(1,0)$ from object to background. This breaks the continuity, since $p_y > q_y > M(x)$, and creates a gap, which results in the following chain of assignments in an upward direction $1 \rightarrow 0 \ldots \rightarrow 1$. As we expect abnormalities of the meniscus to happen only above the meniscus, such a transition should not be permitted. Therefore, the cost of $E_{pq}(1,0)$

should be infinity.

Case 3: Pixels p and q lie above the curve. As in the previous case, assigning the same label should incur a low cost. The cost of $E_{pq}(1,0)$ still does not create a gap, and is the same as $E(0,1)$ from the previous case. A transition $E(0,1)$ from object to background does break the continuity of the segmentation and creates a gap. However, assigning an infinity cost will not permit segmenting abnormalities above the meniscus. Therefore, the choice of cost should allow segmenting folds , but still stick to the boundaries of the meniscus. We notice that in both cases 2 and 3, the cost function is asymmetric: $E(0,1) \neq E(1,0)$.

These three examples and the discussion in section 8.3.1 lead us to the actual choice of costs for $E(1,0)$ and $E(0,1)$. The main observation from these examples is that the difference $|E(1,0) - E(0,1)|$ is what defines the asymmetry. A large difference permits a move in one direction but discourages a move in the opposite direction. In the second case, where pixel q is below the meniscus, we assign $E(1,0) - E(0,1) = \infty$, disallowing a switch from 1 to 0 in an upward direction. Therefore enforcing no discontinuities below the meniscus.

In the third case, where pixels p and q are above the meniscus, the choice of costs is more difficult and depends on the above ratios:

$$c_1 = E(0,1)/E(0,0) \text{ and } c_2 = E(1,0)/E(0,0). \tag{8.6}$$

On one hand, if these ratios are close to (but above) 1.0, the segmentation result will not necessarily be smooth. For a regular meniscus, the segmentation result should be a smooth object with smooth transitions in the segmented height (if needed) as in Figs. 8.9(b) and (d). Low ratios in (8.6) tend to segment a more spiky result. For example, if several vertical pixels on top of the meniscus (which are not part of it) are bright due to illumination error, they might be erroneously segmented.

On the other hand, high ratios (well above 1) in (8.6) might not segment abnormalities (and spikes). If they are segmented, gaps are avoided as the penalties for creating them are high. Fig. 8.3 demonstrates these cases. Firstly, we examine the penalty for segmenting a thin spike as in the right side of the figure. The vertical penalty is built

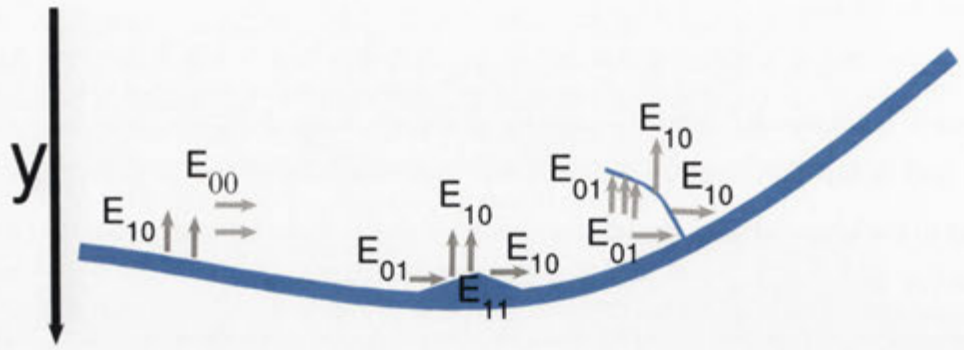


Figure 8.3: *A schematic image of the tear meniscus with abnormalities. In the right side, a thin fold is depicted with the associated costs. In the middle of the figure, a lump is segmented, which can be either a small abnormality that is segmented as a blob or an incorrectly segmented region due to illumination error. The left hand side shows the cost associated with not segmenting anything.*

of the sum: $\sum_k [E(0,1) + E(1,0)]$, where k traverses the perimeter of the spike. To that we have to add the penalty of creating horizontal gaps: $\sum_k [E(0,1) + E(1,0)]$. Secondly, we examine the penalty for segmenting a lump as in the middle of the figure. The main cost is built from the low cost $E(1,1)$ term. Pixels on the outer perimeter of the lump are penalized vertically by $E(1,0)$ and horizontally by $\sum_k [E(0,1) + E(1,0)]$. Thirdly, if no abnormalities are segmented as in the left side of the figure, the penalty is built from the low cost $E(0,0)$ term and from $E(1,0)$ along the outside perimeter of the meniscus. In most cases, not segmenting anything outside of the meniscus borders will trigger the lowest cost if considering only the pairwise term.

Our choice of constants in (8.6) are $c_1 = c_2 = 3$, thus: $E(0,1) = E(1,0) = 3E(0,0) = 3E(1,1)$. This defines the tradeoff between preserving the continuity and allowing to create gaps. The penalty for creating a gap is still relatively high, but can be counterbalanced by the unary term (described next) in case these pixels have to be segmented. We note that if we had prior knowledge regarding the probability of the reservoir being regular or irregular, the choice of parameters could be adjusted.

The rest of the costs for the vertical case and the horizontal case were computed through a learning process. To that end, the upper and lower menisci in 25 images were segmented. We computed the average number of times each of the 4 pairwise terms happen in the horizontal and vertical cases. This was converted to conditional

probabilities defined by (8.5) and used to assign the related costs. For example, the probability of $Pr(\mathbf{x}_p = 1|\mathbf{x}_q = 1)$ for two neighboring vertical pixels in the upper and lower meniscus are 0.85 and 0.79 respectively, which relates to an average meniscus height of 5.5 and 4.5 pixels respectively.

Finally, we show that the cost function can be minimized using graph cuts. The condition is that the submodularity inequality holds (see section 4.5) [65]:

$$E_{pq}(0,0) + E_{pq}(1,1) \leq E_{pq}(0,1) + E_{pq}(1,0). \tag{8.7}$$

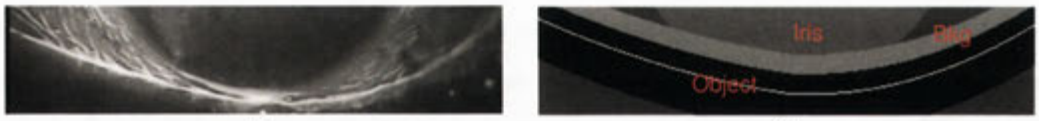
This is the condition that the energy of staying in the same state is not higher then the energy of changing states. As demonstrated, the costs for changing states are high and in particular $E_{pq}(0,1) > E_{pq}(0,0)$ and $E_{pq}(1,0) > E_{pq}(1,1)$, so the submodularity condition holds.

8.3.3 Defining the unary term

The unary term should define a potential function that takes advantage of local image information, such as the pixel's intensity. However, a cost function based only on intensities is not sufficient as it does not utilize the shape prior of the meniscus. Therefore, the unary term combines two independent potential functions:

1. **Gaussian regional model:** Appearance models for the background and the object are built based on pixel intensities. The models are used to distinguish between the (usually) brighter pixels of the meniscus and the background. Additionally, they counterbalance the relatively high cost associated with segmenting folds in the pairwise term. Segmenting these pixels (the folds and abnormalities in general) should incur a low unary cost. We denote this term by $E_i^r(\mathbf{x}_i)$.

2. **Distance constraint:** It is based on the distance of the pixel from the meniscus. Uneven illumination and light coming from different sources make bright pixels artifacts and darker pixels part of the meniscus. These pixels cannot be segmented correctly using only the regional model (since it expects bright pixels to be part of the meniscus and dark pixels to be background). The distance



(a) (b)

Figure 8.4: *(a) Cropped image of the lower meniscus. (b) Object and background seed points. The furthest pixels from the initial spline can be ignored in the segmentation process.*

constraint credits pixels falling near the meniscus regardless of their intensity. It is denoted by $E_i^d(\mathbf{x}_i)$.

Therefore, the unary term is defined as the sum of two independent potential functions as follows:

$$E_i(\mathbf{x}) = E_i^r(\mathbf{x}_i) + E_i^d(\mathbf{x}_i). \tag{8.8}$$

8.3.4 Regional model

The regional model is estimated by first finding seed points that belong to the object (tear meniscus) and the background. These points are then used to build a probability distribution function for the object and background.

Seed points detection

Given the original image $I(x, y)$ and θ_m, the image is cropped, so the segmentation is focused on a small area (see Fig. 8.4(a)). From now on, we only consider the cropped image, which will be denoted as $I(x, y)$ as well. For the object seeds, we choose all seeds that are on the interpolated spline (see section 8.2). In most cases, these seeds are brighter than the background. We note that in some regions, the tear meniscus height can be one pixel only. If the initial B-spline misses the meniscus by even one pixel, the dependency of the potential functions on the shape prior can result in an incorrect segmentation. Usually there is no simple way to know that unless there is a large decrease in intensity and in such a case, these pixels can be disregarded.

Detecting background seeds is easy. However, it is of interest to find those that are closest to the meniscus in the spatial domain. In section 5.3, we showed how the iris

can be detected in our images as a circle. The iris can be used for background seeds, but since it is generally much darker than the meniscus, it is just marked and it is not used for estimating the background model. Pixels on the conjunctiva are usually closer in intensity to the meniscus and can be segmented as background if they are far enough from the estimated meniscus. This distance is computed by correlating to the clinically observed max height of the meniscus in most individuals [39] (and we add an error margin). In our images we define $h_l = 14$ and $h_u = 18$ as the maximum height in pixels of the lower and upper menisci respectively. For a pixel $i \in \mathcal{P}$ with coordinates (x, y), we define:

$$d_y(i) = |y - M(x)| \tag{8.9}$$

which is the vertical distance of the pixel from the B-spline ($M(x)$ was defined in (8.2)). A background pixel is taken when: $h_l \leq d_y \leq 2h_l$. We note that some pixels might be part of the iris, but usually only a small portion. Finally, all other pixels such that $d_y > 2h_l$ are ignored in the segmentation process. An example of the detected seeds is given in Fig. 8.4. The background and object seeds are not forced to be segmented as such, but are just used to build the model. We note that the seed points are only used to build the model for the unary term and are not needed for the initialization of the graph-cut algorithm.

Likelihood function

After detecting seed points that belong to the object and background, we use this information to create probability distribution models based on pixel intensity. For a given pixel i with intensity I_i we simply would like to compute the following log likelihood function: $Pr(I_i|x_i)$. This is the probability that pixel i belongs to the data model, given it is assigned the label x_i. In other words, it defines the penalty associated with assigning pixel i with label x_i. The use of log likelihood functions for regional terms has appeared before and is motivated by [16].

Learning from given examples, we found that in most cases, it is adequate to define the object and background models using a Gaussian distribution. We compute

the mean and standard deviation of the background seeds: $\Sigma_0 = (m_0, \sigma_0)$, and similarly for the object seeds: $\Sigma_1 = (m_1, \sigma_1)$. The following 1-D Gaussian functions for the background and the object are defined as a function of the pixel's intensity t:

$$\begin{aligned} G_0(t; \Sigma_0) &= \mathcal{N}(t; m_0, \sigma_0), \\ G_1(t; \Sigma_1) &= \mathcal{N}(t; m_1, \sigma_1) \end{aligned} \tag{8.10}$$

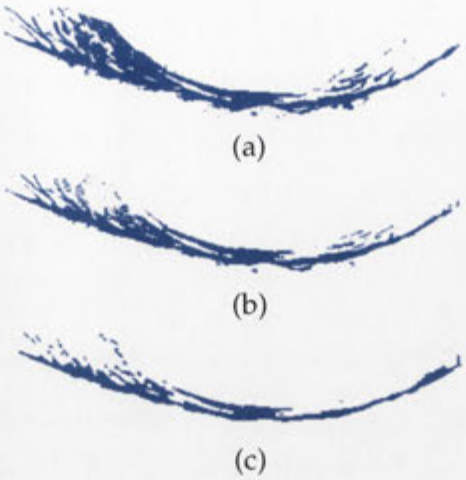
For each pixel $i \in \mathcal{P}$ ($\mathcal{P}$ is the set of pixels in the image) with intensity I_i we compute the log likelihood of it belonging to object and background. This is the conditional probability of classifying pixel i into given intensity models of the object and background:

$$\begin{aligned} E_i^r(\mathbf{x}_i = 0) &= -\log Pr(I_i|\Sigma_0), \\ E_i^r(\mathbf{x}_i = 1) &= -\log Pr(I_i|\Sigma_1) \end{aligned} \tag{8.11}$$

In (8.11), the higher the probability of the pixel fitting into the given model, the lower is the penalty given by E_i^r. Fig. 8.5(a) shows the segmentation result of the meniscus from Fig. 8.4(a) when using only the regional model for the segmentation. Most of the meniscus pixels have been segmented, but also pixels outside the meniscus, which do not belong to the object.

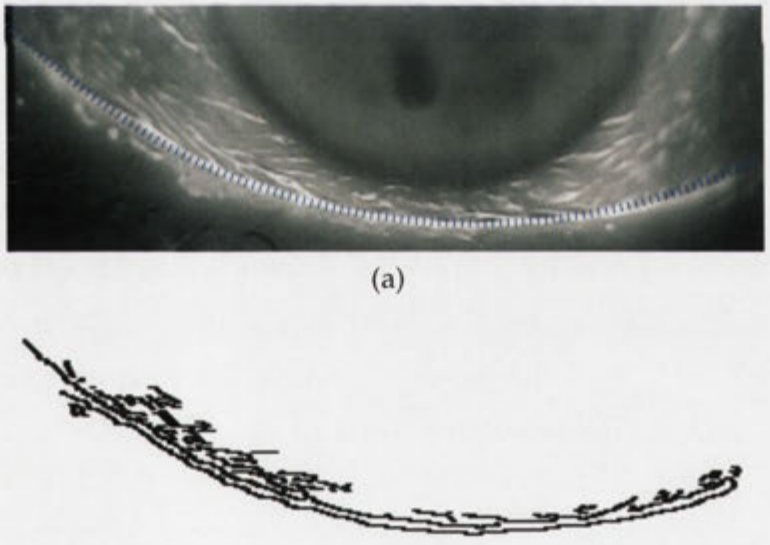
8.3.5 Distance constraint

The distance constraint incorporates information regarding the shape and the spatial location of the meniscus. The idea is that given the meniscus shape prior and the average height of it in different spatial locations, we can estimate the probability of labelling pixels above and below it as object. For example, let us assume pixel p with coordinates $p = (p_x, p_y)$. Then $M(p_x)$ is the y coordinate of the shape prior at the same x coordinate as p. The value $|p_y - M(p_x)|$ determines the vertical distance of the pixel from the shape prior. The larger the difference, the higher should be the penalty of labelling p as object. The exact penalty should be a function of this distance and the average height of the meniscus at that spatial location.



(a) (b) (c)

Figure 8.5: *Segmentation of the meniscus of Fig. 8.4(a) when using: (a) only the regional potential function of the unary term (b) both potential functions of the unary term. It is evident that the distance constraint cuts most of the pixels falling far from the curve. The pixels that are still segmented are mostly bright pixels falling on edges. (c) Final segmentation result using both unary and pairwise terms*



(a) (b)

Figure 8.6: *Explaining the distance constraint. (a) Cropped image of the lower meniscus with the initial estimation of the B-Spline dotted in blue. (b) Edge information where only the pixels that are connected to the meniscus are shown. All these pixels will be assigned a zero distance in* $E_i^d(x_i)$

The distance $d_y(i)$ of a pixel i from the curve is defined as in (8.9). However, pixels that belong to branches such as in figures 8.4(a) and 8.6(a) which originate from the meniscus curve should not be heavily penalized. These pixels can have a large vertical distance from the meniscus curve, but they are still part of the meniscus and should be labelled as object. These branches usually have salient features and are evident in the edge map of the image. We would like to include this information in the distance constraint function.

To that end, we find all 8-connected regions in the (cropped) image and denote the set of all pixels of the k-th region by B_k. If the connected region originates from the meniscus (for example a fold), it should be connected to it at some point. All pixels that belong to that region should not be penalized. For each connected region, the minimum distance to the shape prior is computed. In another words, we find the vertical distance from the pixel closest to the shape prior to the shape prior. If this distance is close to 0, we assume that the pixels of this region are part of an irregular meniscus and do not penalize them. We define a new distance function d_y^b, such that:

$$d_y^b(i) = \begin{cases} 0 & \text{if } i \in B_k \text{ and } \min_{i \in B_k} d_y(i) \leq 1 \\ d_y(i) & \text{otherwise} \end{cases}$$

Therefore, if the distance of the closest pixel in the region to the shape prior is less than 1, all pixels in that region (see Fig. 8.6(b)) are assigned a distance of 0. These pixels have a higher probability of being an abnormality and should be segmented, thus assigned $d_y^b(i) = 0$ (no penalty). Finally, we use the same likelihood function as in [61] for the CRF: $E_i^d(\mathbf{x}_i) = -\log Pr(\mathbf{x}_i|\theta_m)$, where θ_m are the set of parameters that define the shape prior. This probability is defined as:

$$Pr(\mathbf{x}_i = 1|\theta_m) = 1 - Pr(\mathbf{x}_i = 0|\theta_m) = \frac{1}{1 + \exp(d_y^b(i) - d_r)} \tag{8.12}$$

The parameter d_r defines the average height of the meniscus and its value can be learnt by using the average height of the meniscus (when excluding the abnormalities). Also this value can vary for different spatial locations of the meniscus. For example, we might have a small value of d_r in the nasal and temporal regions, and a higher value in

the central region. The improved segmentation when using both potential functions of the unary term is depicted in Fig. 8.5(b) and the final segmentation result when using both the unary and pairwise terms is in Fig. 8.5(c).

8.4 Minimizing the function

The CRF of (8.3) depends on the eyelid shape prior through its set of parameters θ_m. Thus, the cost is dependent on both the labelling and the parameters θ_m. The minimization problem is defined as finding the least energy labelling with the best shape prior:

$$\min_{\mathbf{x}} E(\mathbf{x}) = \min_{\mathbf{x};\theta_m} E_i(\mathbf{x}_i;\theta_m) + E_{ij}(\mathbf{x}_i,\mathbf{x}_j;\theta_m) \tag{8.13}$$

The least cost is not always obtained with the initial estimation of the shape prior. For example, in Fig. 8.6(a) the estimation of the meniscus is dotted in blue and misses the correct meniscus by a few pixels near the right side due to lack of edge information. As the pairwise term penalizes for discontinuities, the meniscus region below the initial curve might not be segmented at all or the whole area between the estimated curve and the meniscus will be segmented. Both segmentations are incorrect. Thus, a process that updates the curve parameters is needed.

The graph-cut algorithm is used to solve a binary problem and the optimum is guaranteed to be found. However, our cost function of (8.13) is a CRF potential function which is based on the shape prior in both the unary and pairwise terms. Therefore, changing the shape prior can result in a different optimum, thus the least cost is not necessarily achieved with the initial shape prior. As we do not know the best shape prior initially, we use an iterative process to solve the problem. Since the underlying data changes (θ_m), the value of the potential functions changes and hence the graph-cut algorithm produces a different segmentation result.

We note the iterative process is required since we minimize a CRF based energy function. Even though the underlying image does not change, the distance function (in the unary term) and the pairwise term have to be reevaluated when the shape prior changes. This is a significant distinction that has to be made clear when compared to

the use of a MRF based energy function.

8.4.1 Algorithm

We are now ready to present a pseudo-code algorithm to segment the meniscus. The algorithm is an iterative one. In the heart of the algorithm lies the graph cut minimization process as was described in previous sections. The output of the graph-cut algorithm is a binary segmentation. This segmentation is the optimal one for a given cost function and shape prior. However, as discussed before, the initial shape prior is not necessarily correct and so is the meniscus segmentation induced by it. By iteratively updating the location of the knots of the shape prior, we obtain new segmentation results (using the graph-cut algorithm). Updating the knots is carried out using the Levenberg-Marquardt (LM) minimization algorithm (detailed below) and the min-cut is computed at each iteration of the LM algorithm. Finally, at times the initial shape prior significantly differs from the correct one (at certain spatial locations). To overcome that, we manually adjust the knots of the shape prior and repeat the minimization process using the LM algorithm. Therefore, the segmentation result does not utterly depend on the initial shape prior and a correct result is obtained through iteratively updating its parameters.

Algorithm *Segment Meniscus*

Input: Initial shape prior θ_m

Output: Segmentation result and the best shape prior

1. Find the initial shape prior parameterized by θ_m ($*$ see section 8.2 $*$)
2. $(bestCost, bestSegmentation) \leftarrow solveUsingGC(\theta_m)$ ($*$ Obtain initial segmentation $*$)
3. $currentPrior \leftarrow initialShapePrior$ ($*$ parameterized by θ_m $*$)
4. $bestPrior \leftarrow initialShapePrior$ ($*$ parameterized by θ_m $*$)
5. ($*$ Iterate over the spline knots $*$)
6. **for** $knotsNumber \leftarrow 1$ **to** $numberOfSplineKnots$
7. ($*$ Iteratively obtain segmentation results by updating the current prior using the LM algorithm and running the graph-cut algorithm using the

new prior $*$)

8. $(cost, prior, segmentation) \leftarrow SegmentUsingLM(currentPrior)$
9. **if** $cost < bestCost$ ($*$ A better segmentation was found $*$)
10. **then** update bestCost, bestPrior and bestSegmentation
11. $currentPrior \leftarrow bestPrior$ ($*$ Use best prior for the next iterations $*$)
12. update spline knot $knotsNumber$ of currentPrior
13. **return** bestSegmentation and bestPrior

Our two step minimization algorithm is summarized by the algorithm *Segment Meniscus*. In line 2, an initial segmentation result is found using the initial shape prior. Then starts the iterative process, where the shape prior is updated at each iteration: In the outer loop (lines 6-12), spline knots are explicitly modified; In the inner loop, the iterative Levenberg-Marquardt (LM) algorithm [41] is called (line 8) and implicitly updates the shape prior at each iteration.

The LM algorithm (line 8) is used to iteratively refine the estimation of the meniscus curve and the associated min-cut cost. At each iteration, a call to the graph-cut algorithm is made (as in Line 2) with the current shape prior. The input vector to the LM algorithm is built of the knots of the current B-spline (see section 8.2) and the output is the segmentation result and the min-cut cost. The B-spline knots are adjusted at each iteration of the LM algorithm to create a new shape prior (to be used by the graph-cut algorithm). The min-cut cost is used by the LM algorithm to estimate the next spline parameters until (the guaranteed) convergence.

In some cases, the LM algorithm only finds a local minimum and not the global one. This can happen if the initial estimation of the shape prior is incorrect. For example, in a badly illuminated image, the eyelids might have weak edges near the sides of the image and the error in estimating the shape prior can be relatively large. Thus, the LM algorithm will not necessarily find the correct solution.

These cases are handled by the outer loop, where individual spline knots are explicitly modified at line 12. The updated shape prior is used as the current shape prior for the next iteration of the inner loop. If the LM algorithm produces a cost which is lower than the best one (line 9), the segmentation result, the cost and the shape prior

are saved (line 10). Further iterations will use this best shape prior (line 11). Finally, the least energy obtained defines the best labelling and hence the segmentation result and the best shape prior.

Our segmentation algorithm is close to the PoseCut algorithm [61] mentioned in the previous work section. In that paper, the authors use the graph-cut minimization algorithm to simultaneously segment a human from an image and estimate its pose. Their cost function is a CRF-based energy function that depends on a set of parameters that define the shape prior (a stick-man model). The initial shape prior is an educated guess of the human pose (for example, its pose in the previous video frame). Throughout the iterative process, its parameters are updated. Updating the parameters of the human pose is done using the Powell minimization algorithm. Even though the methods share similar ideas (such as the distance constraint), we present a method that uses an asymmetric pairwise cost function, which produces a segmentation result that follows a particular shape induced by the shape prior. In [61], the pairwise term is symmetric and the similarity measurement to the pose shape prior (the stick-man model) is carried out in the unary term.

Fig. 8.7(a) demonstrates the initial shape prior found after the B-spline initialization (with a min-cut cost of 7.18179e+006). In the left side, it missed the meniscus curve (due to weak edges). After the first iteration of the LM algorithm the shape prior is updated (with a min-cut cost of 7.14519e+006) to the one in Fig. 8.7(b). In the third iteration of the LM algorithm, the min-cut cost is reduced slightly more (7.14117e+006) and the shape prior is updated again. However, the change in the spline knots is very small, so it is not drawn. The associated cost is not reduced in further iterations.

8.4.2 Improving the computation time

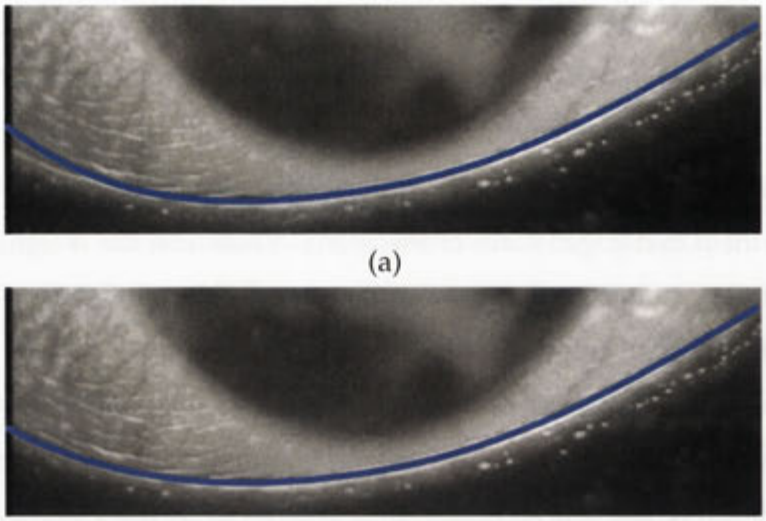
Our iterative minimization algorithm relies on a parameterized underlying model θ_m, where the best parameters are unknown. Finding the best segmentation and the best parameters requires changing the parameters of the model at each iteration. Thus the unary term or the pairwise term (or both) of the cost function change as well. This results in new weights (capacities) for the graph's edges and possibly in a different

max-flow. Quite often, the new min-cut graph (lowest energy) is very similar to the previous graph, since not many costs are changed. For example, changing the location of one knot of the B-spline affects the distance constraint of only those pixels in the realm of this knot. The pairwise term only changes for those pairs of pixels that at least one of them exchanged sides of the spline. Therefore, the weights assigned to the edges of the graph from the previous iteration will be similar for a large portion of the nodes in the new graph.

Kohli and Torr in [62] present a new method called dynamic graph-cuts (see section 4.6 for more details) to accelerate the computation time of a very similar problem. At each iteration of the LM algorithm in line 8, a call to the graph-cuts algorithm (same as max-flow) is made with different parameters for the shape prior. However, the change to the parameters is usually only small. Thus, dynamic graph-cuts is an appropriate solution to speed up the computation time. Instead of creating a new graph at each iteration and computing the max-flow algorithm, the previous graph is reused. The number of edge weights that are changed is proportional to the number of pixels that their cost function changed. Then, the new flow is computed by only searching for augmenting paths that go through pixels with a cost change. The additional flow through these nodes is limited to the additional capacity in this nodes. Therefore, the number of augmenting paths is bounded by the sum of the total increase in capacity.

8.5 Results

A qualified clinician segmented the upper and lower menisci in 25 images, resulting in a total of 43 segmentations (in some the upper meniscus is covered by eyelashes). All images were taken from the recorded videos and are of resolution 352×288. Fig. 8.8 shows segmentation results using simpler models: In the first row only a threshold is used and the results are useless even when choosing the best thresholds. In the second row, we used the same unary term and a Potts model [17] for the pairwise term (symmetric). Small areas below the meniscus were wrongly segmented and also the result is slightly less smooth. Fig. 8.9 shows original meniscus images (cropped) with their segmentation result. Figures 8.11 and 8.12 show the manual and automatic



(a)

(b)

Figure 8.7: *Demonstrating the iterative minimization process. (a) The initial shape prior found in section 8.2. The shape prior mostly follows the meniscus curve, except for the left side. (b) The improved shape prior after the first iteration of the LM.*

segmentation results of regular and irregular menisci along with the best optimized prior.

We compared the number of matching pixels - true positives (TP), and the number of incorrectly segmented pixels - false positives (FP). The FP were computed as a percentage from the cropped image area (see first row of Fig. 8.9), and not from the whole image (which would have made the FP lower). The upper and lower menisci had on average 1690 and 1603 segmented pixels out of 25562 and 35066 pixels respectively. For the lower meniscus, the TP is 82.7% and the FP is 4.8% and similarly for the upper meniscus, TP is 88.0% and FP is 3.6%. These results provide an idea of the quality of the segmentation and are on the accepted range.

However, no data related to the meniscus structure is given so far. A grading of the quality of the meniscus was defined by a clinician as described in table 8.1. The clinician graded the meniscus at all images and provided a definition of thick and thin meniscus. Using the segmentation result, we employed an automatic grading technique: The average height of the meniscus is computed over the central, nasal and temporal areas (using the estimated location of the iris) and is used for assigning

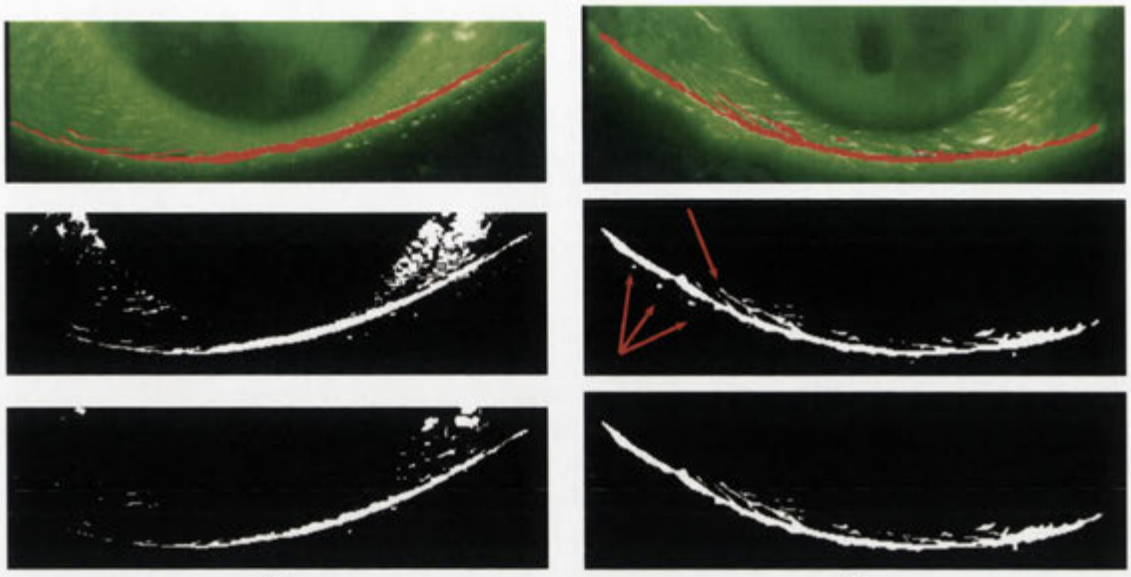


Figure 8.8: ***Column (a) from top*** *- segmentation result when using only a threshold: Manually segmented image for lower meniscus; Threshold image: TP =74%,FP=5%. ; Threshold image when using a slightly higher threshold. TP=61%,FP=1.5%. Our segmentation results is in Fig. 8.1(b) TP=85%,FP=1.5%.* ***Column (b) from top*** *- Manually segmented image for lower meniscus. ; Segmentation result when using our unary term and Potts model for the pairwise term - Small areas below the meniscus are segmented and also one of the lines is incomplete. ; Our segmentation result using both unary and pairwise terms.*

a grading of 1 to 3 (these gradings are mainly dependant on change of height of the meniscus through its length).

Folds (grading 4) are detected by treating the segmentation result as a tree. A thinned image of the meniscus is created using medial axis representation. The connected part that spans the image from left to right is considered as the tree root. the tree is delineated using the best shape prior (8.13). Any other objects are considered as branches. Each tree branch with a length over a threshold is related in length and location to an actual folding. Therefore, the height of the meniscus and number of branches define the automatic grading. For example, the image in Fig. 8.9(g) was graded 4, as it had one long branch (fold) and an average height of 5.2 & 6.8 pixels over the conjunctiva and the cornea respectively. Figures 8.11 and 8.12 show the tree skeleton that is created from the segmented image. The main curve of the meniscus is plotted in blue and any branches are in red.

Grading	Description	Figure
1	**Regular** - Thick in corneal and conjunctival areas	8.9(a)
2	**Regular** - Thin in corneal and conjunctival areas	8.9(c)
3	**Irregular** - Thick corneal area; thin conjunctival area	8.1(a)
4	**Irregular** - Branching and/or folding exist	8.9(g)

Table 8.1: *Defining the grading of the meniscus using 4 categories.* ***Thick*** *and* ***Thin*** *refer to the height of the meniscus. The last column of the table is a reference to a figure showing a meniscus having this grading (according to the clinician).*

Grading	Description	Detection	Description of Error
1	**Regular**	6/8	Grading '3' was detected twice
2	**Regular**	2/3	Grading '3' was detected
3	**Irregular**	2/5	Folds were detected twice
4	**Irregular**	9/9	

Table 8.2: *Comparing the automatic grading of the meniscus to the clinician's grading. The third column shows how many gradings were detected correctly. The last column describes what grade was detected instead.*

We compared the automatic and manual gradings of the lower menisci (all but 1 upper menisci were regular) and present the results in table 8.2. We notice that all irregularities containing folds (or branching) were discovered correctly by our algorithm. Also most of the regular cases were detected correctly. Differentiating between the categories is a hard task even for the clinician and categorizing the meniscus as irregular (category 3) and not regular can be sometimes subjective. The grading can be better refined through clinical trials, such as allowing a multiple category grading or providing the degree of folding.

8.6 Summary and further research

We have presented a new automatic method to segment the tear meniscus in eye images and analyze its quality. This analysis is given to the clinician in conjunction with the analysis of dry areas. Together, they provide a more robust clinical evaluation of the dryness symptoms and origin to the clinician.

Our method is based on minimizing a quadratic cost function, where both the unary and pairwise terms use an underlying model to determine the cost. The main

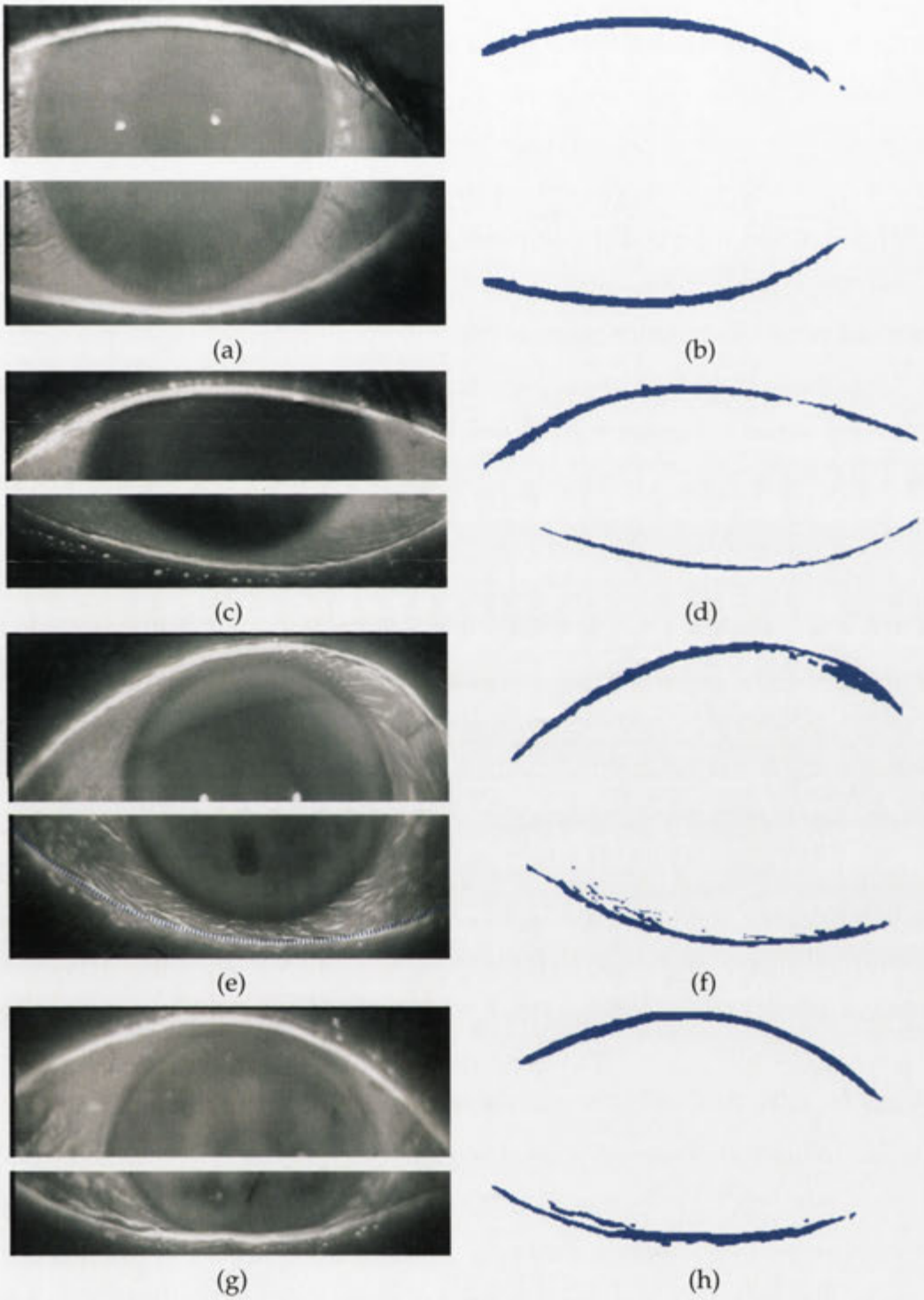


Figure 8.9: *Results. (a) Healthy meniscus with a constant height. (c) Healthy meniscus with a constant thin height. (e) Irregular thin meniscus with a few branches. (g) Irregular meniscus with an evident folding and a varying thickness.*

novelty is in the pairwise term, which introduces asymmetry in order to add a sense of direction to the segmentation process: movements in one direction are permitted but not always in the opposite direction. The minimum of the function is found using an iterative minimization algorithm, where the estimation of the meniscus curve (or shape prior) is being updated at each iteration, thus affecting the computation of the cost terms. Once converged, the output is the segmentation result with the best shape prior. Therefore, even if the initial meniscus curve is not estimated correctly, it will be adjusted during the iterative process.

After the segmentation process, the output image is analyzed to provide the clinician with automatic feedback about the shape of the meniscus. If it is irregular, the folds are highlighted. This information is recorded and can be compared at a later time.

The main challenge of the method is to define the tradeoff between sticking to the boundaries of the meniscus and segmenting abnormalities, thus creating gaps in the segmented object. The solution we have presented in this chapter does segment these abnormalities by: (1) moderating the penalty of the pairwise term for creating gaps and (2) using the edge image to find structures that originate from the meniscus and not penalize them in the unary term.

One immediate extension to the method is to measure the tear meniscus height at different times and not only after two seconds. Even though the tear formation stabilizes after two seconds, it is of interest to see if any changes happen in the tear reservoir height and shape throughout time. If changes occur, they can be related to some irregularity in the tears or suggest a better timing to evaluate the meniscus. Clearly it can be done easily by our algorithm, requiring only to run the algorithm with different input images at different times and compare the results. A graph of the tear meniscus height can be created over time. The same task can be very tedious for a clinician especially on a large database.

Even though our solution performs well in most cases, it still suffers from a few drawbacks: (1) The model used in the regional term cannot always depict correctly the varying intensity of the meniscus due to illumination problems (see second row of Fig. 8.10). (2) Bright artifacts towards the ends of the meniscus can produce segmented

blobs that stick out from the meniscus boundaries (see first row of Fig. 8.10). (3) Images with reflex tearing produce segmentation results which are either too thick or too thin (see third row of Fig. 8.10). Reflex tearing can be best handled by not segmenting these images at all as they do not depict the real shape of the meniscus. Repeating the clinical routine to achieve an image without the tearing is preferable. To summarize, our method is not flawless but these case are not common and some of the illumination errors can be better controlled in the clinic.

Finally, the automatic grading has to be revised as we gather more clinical data. In our experiments, we have used the clinician's grading as the gold standard. Although we believe that a grading of 4 categories is enough, the differences between the categories are not clear-cut. One solution is to allow the clinician to combine gradings. For example, a meniscus can have a grading 1 (regular), but a grading 3 in the temporal area (change of thickness). Clinical trials to evaluate intra and inter observer differences can provide a better idea on how the height measurement varies between clinicians and where our measurement stands. In addition, the repeatability of the gold-standard between clinicians could be estimated better.

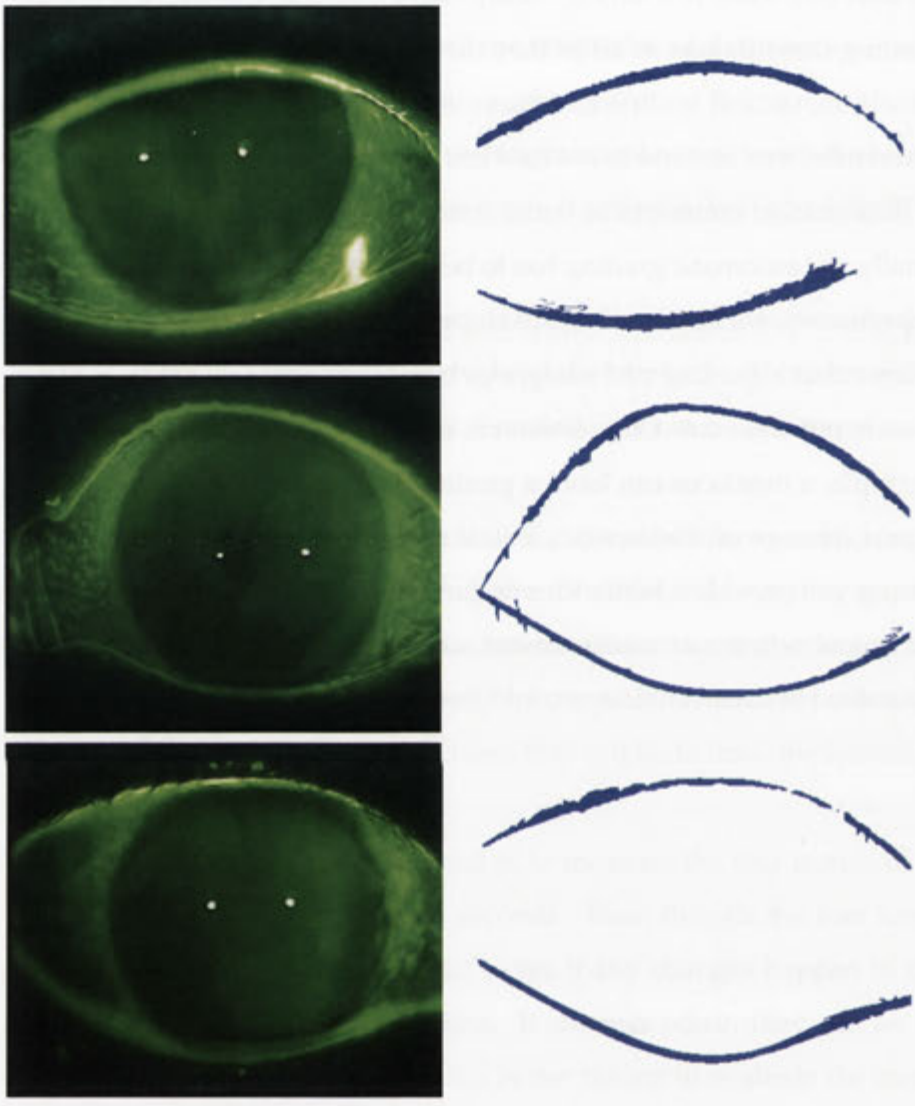


Figure 8.10: *Images with poor illumination resulting in segmentation errors.* ***First row:*** *Parts of the badly illuminated big blob in the right hand side are segmented. It resulted in a segmentation which is thicker than needed.* ***Second row:*** *Both the lower and upper menisci are much darker in the center than in the sides. In addition, some areas on the conjunctiva are brighter than the meniscus. The segmentation result captures the general shape of the meniscus, but fails to segment its whole height in the center.* ***Third row:*** *Reflex tearing hinders the correct segmentation.*

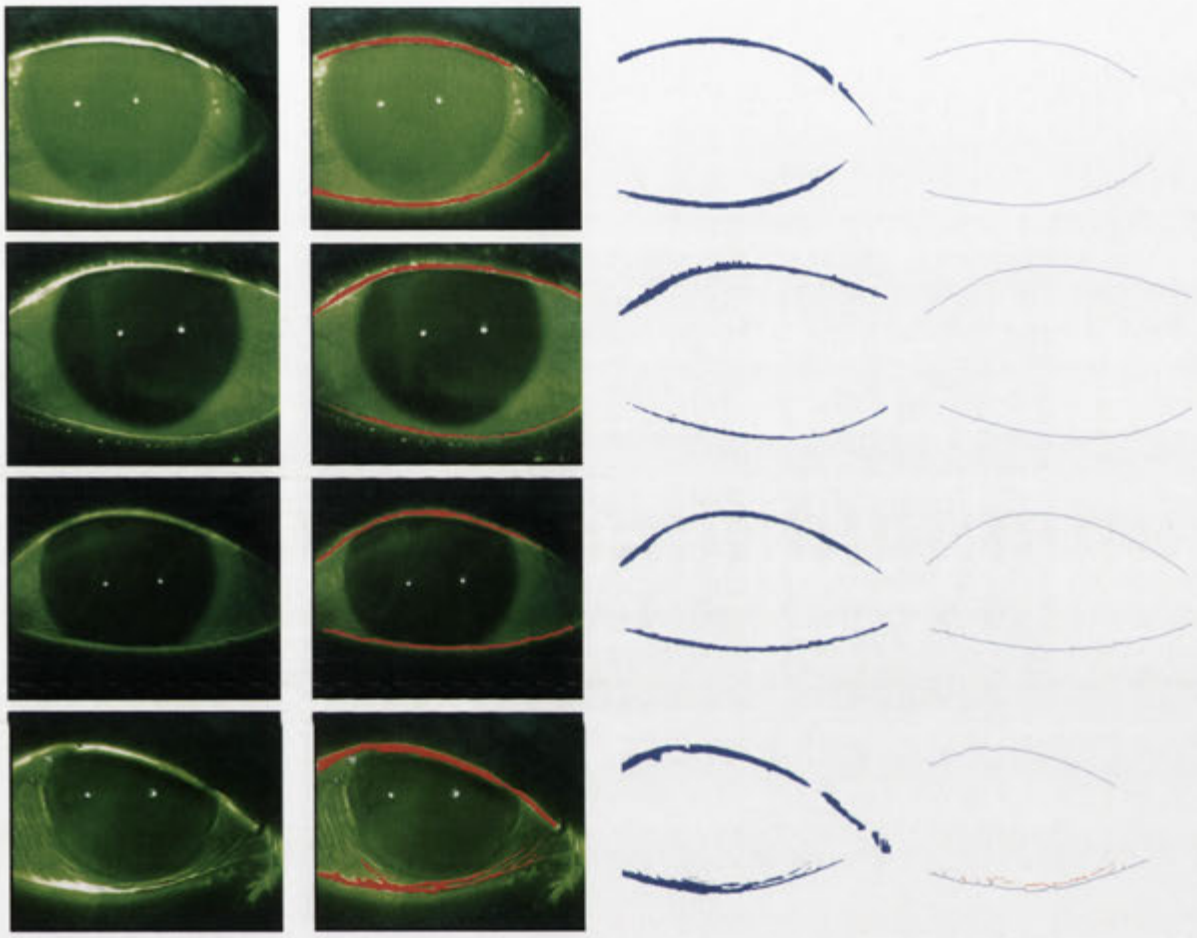


Figure 8.11: *Results using our algorithm demonstrating the four possible gradings of the meniscus. First column: original image. Second column: manually segmented meniscus. Third column: automatic segmentation result. Fourth column: the tree skeleton that is built for the automatic grading (see description at the results section). The blue pixels are the main tear meniscus and the red pixels are branches. Branches over a certain length are considered to be folds and hence grading 4.* ***First Row:*** *Regular meniscus grading 1.* ***Second Row:*** *Regular meniscus grading 2.* ***Third Row:*** *Irregular meniscus grading 3.* ***Fourth Row:*** *Irregular meniscus grading 4. Due to the asymmetric pairwise term, the best shape prior follows the lower part of the meniscus, while any abnormalities appear above it.*

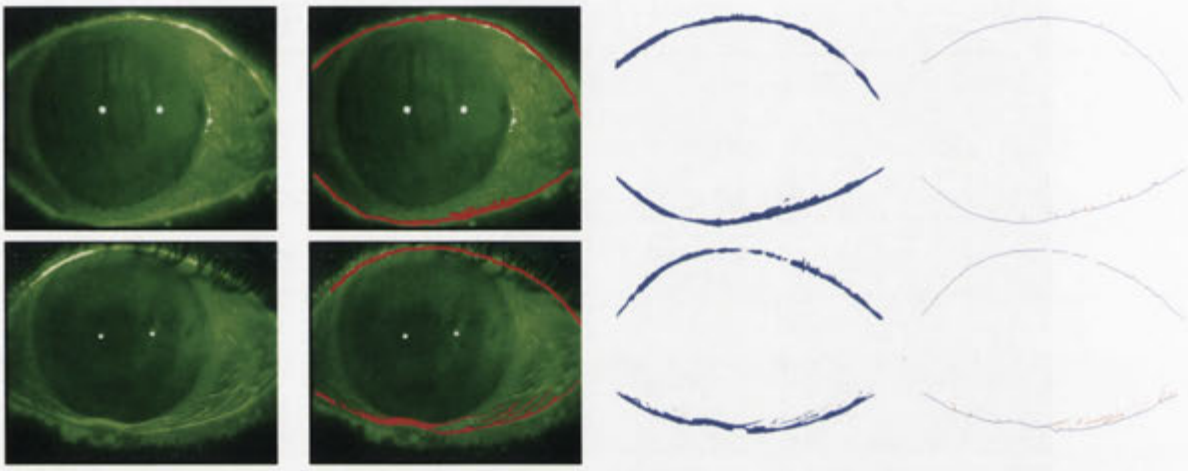


Figure 8.12: *Two more challenging images. First column: original image. Second column: manually segmented meniscus. Third column: automatic segmentation result. Fourth column: the tree skeleton that is built for the automatic grading (see description at the results section). The blue pixels are the main tear meniscus and the red pixels are branches. Branches over a certain length are considered to be folds and hence grading 4.* ***First Row:*** *Irregular meniscus grading 3. Notice the big differences in intensity in the upper meniscus and the change of heights in the lower meniscus.* ***Second Row:*** *Highly irregular meniscus grading 4. Notice, that modelling the meniscus with a spline is not an easy task as the branching is quite prominent.*

Chapter 9

Conclusions and discussions

The main focus of this thesis is to study the dry eye problem and present an algorithm to automate the detection of the disease. The aim is to build a comprehensive system that includes the analysis of a number of dryness symptoms (such as the BUT, the black line and tear meniscus evaluation).

We decide to approach the problem by using fluorescein imagery, which is used by clinicians to perform the Fluorescein Break Up Time (FBUT) test. This is probably the most commonly used test by clinicians and is very accessible as the required equipment can be found in almost every clinic. In addition, we record our videos using a hand-held camera, thus increasing the accessibility of our solution.

We define and implement an automatic algorithm to detect dryness symptoms in fluorescein videos. Our method is based on three crucial steps: detection of the iris and the eyelids, alignment of the video and segmentation of regions of interest. We were able to segment the dry regions and provide a detailed clinical analysis of dryness symptoms such as: the various degrees of tear film thinning, the shape and size of the dry regions and the progress of the break regions. All these symptoms are reported to the clinician in a graphical way. At most times, this analysis provides much more information than the clinician is capable of observing when performing the FBUT test. To our knowledge, this is the first time such an automatic system has been built and the accompanied experiments have been performed. Our results show that dryness symptoms can be reliably detected in real-time and suggest the cause of the dryness.

We have carried out experimental tests to estimate the inter-observer and intra-observer variance among clinicians. Four clinicians were given a set of videos and

were instructed to estimate the Break Up Time (BUT). The results were surprising: low intra-observer variance but high (up to very high) inter-observer variance (the difference in BUT estimate between clinicians). These results corroborate the need for a more robust method to estimate the BUT as was suggested by the International Dry Eye Workshop (DEWS). To that end, we define a new value to measure the BUT. We suggest a value based on combining the clinical definition of a break and spatial and temporal image properties. We call this value Digital Electronic Break Up Time (DEBUT), and explain how it is computed. The main advantage of the DEBUT over the BUT lies in its automatic estimation and thus couples two key benefits: non-dependency on the operator and reproducibility. We also show that the DEBUT is in agreement with the BUT values computed by the clinicians.

The results of the tests suggest that the scope for further work is still open both in the clinical and medical-imaging fields. First, it is important to define clear guidelines for the clinicians to follow when analyzing the recorded videos. Second, it is unclear if all clinicians define the BUT in a same way: for example, does the break happen when an area is becoming dark or when it is not getting any darker? Third, data should be collected over time with the same dry eye patients to corroborate and improve the repeatability of the algorithm. Finally, the results from the tests should be implemented in the automatic algorithm to refine the computation of the DEBUT value.

A multi-label graph-cut algorithm based on alpha-expansion is presented in chapter 6 to improve the detection of dryness. The extension is shown to produce more robust results, which are less sensitive to noise and movements of the iris and the camera. This is due to the introduction of the monotonic constraint which fits perfectly with the definition of tear film instability. The graph-cut solution enforces spatial smoothness which agrees with the local pattern of dry regions. In addition, a more sensitive threshold is used for the computation of the DEBUT. Thus, break areas are detected even if they do not increase in size (over time) and a more accurate timing is provided for videos with breaks that have slow progress. This method gives the most accurate estimate of the DEBUT when compared with the clinicians.

A graph-cut algorithm for evaluating the tear meniscus height and shape is presented in chapter 8. We demonstrate how an asymmetric cost function and an itera-

tive multi-step optimization method find simultaneously the best shape prior of the meniscus and the segmentation result. The segmentation result is analyzed for its shape and categorized into regular and irregular. We show that good segmentation results are achieved and irregularities such as folds are detected.

Future research related to the tear meniscus can be carried out. From a medical point of view, data should be collected to understand better the agreement between its height and dry eye. From an engineering point of view, machine learning algorithms can introduce additional heuristics to the optimization method. For example, the learning process can determine the next shape prior throughout the iterative process. Such heuristics can reduce the number of iterations and the computation time.

Evaluating the tear meniscus, segmenting the dry areas, detecting the black line and computing the DEBUT value are all computed by using the same input video. By combining the results of the algorithms, we create a comprehensive system that addresses different causes of the dry eye problem. For example, a patient will not always show symptoms of dryness in the FBUT test, but can have a shallow or irregular tear meniscus related to a thin tear film aqueous layer. Therefore, the origin of the dryness can be attributed to the deficit of fluid. Another example is a complaining patient with a regular tear meniscus, no break up time, but with symptoms related to the black line near the lower eyelids. The dryness might be attributed to improper blinking or to thinning along the lid margins. Such a system can be installed in the clinic and provide immediate results to the clinician, as performing the FBUT test and running our algorithm only require a couple of minutes.

An unsolved problem is the detection of staining, a more severe form of dryness. Staining can be detected using the existing fluorescein videos and improve the knowledge provided by the system. This requires collecting data from staining patients, which are less prevalent compared to dry eye patients.

Appendix A

Tracking of Blood Vessels in Retinal Images

We present an automatic method to segment the blood vessels in retinal images. Our method is based on tracking the center of the vessels using the Kalman filter. We define a linear model to track the blood vessels, suitable for both the detection of wide and thin vessels in noisy images. The estimation of the next state is computed by using gradient information, histogram of the orientations and the expected structure of a vessel. Seed points are detected by a set of matched filters in different widths and orientations. Tracking is carried out for all detected seed points, however we retrace the segmentation for seeds with small confidence. Our algorithm also handles branching points by proceeding in the previous moving direction when no dominant gradient information is available. The method is tested on the public DRIVE database [103] and shows good results with a low false positive rate.

A.1 Fundus imaging

In fundus images the interior surface of the eye is being photographed and includes the retina, optic disc and macula. A typical camera provides views of 30 to 50 degrees of the retinal area and produces color images. Usually the clinician takes several images of the different areas of the retina. The patient is asked to move his gaze to follow an external fixation light and as a result the eyeball moves and rotates, resulting in a greater field of view. Angiograms, black and white images, can be produced by injecting a dye and waiting a few minutes for it to spread. The advantage of angiograms is

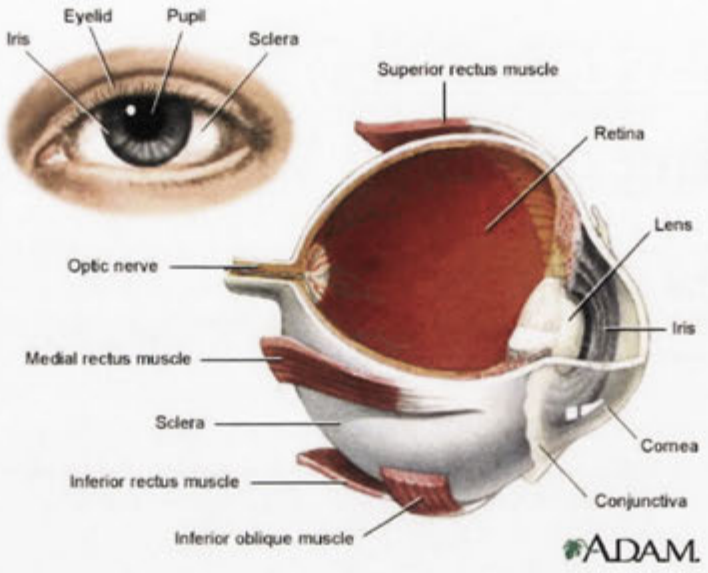


Figure A.1: *Cross-section of the eyeball, showing the relationship between the iris, cornea and the retina. The retina is the inside layer, which includes the optic nerve and the macula. Courtesy of ADAM http://adam.about.com/encylopedia*

that some lesions are much more easily detected (such as microaneurysms). The disadvantage is that it is an invasive method, and some patients might not react well to the instillation of the dye. More information about the eye structure and fundus photography can be found in [112]. It is also clear that fundus images are very different from images of the anterior of the eye discussed so far in this work.

A.1.1 The retina

Light enters through the pupil, is focused and inverted by the cornea and lens, and is projected onto the back of the eye. At the back of the eye lies the retina. The optics of the eye create an image of the visual world on the retina, which serves much the same function as the film in a camera. Stimulated by light that reaches the retina, the iris expands and contracts to enlarge or reduce the size of the pupil. The automatic exposure mechanism adjusts the amount of light that falls on the retina.

The neurons in the retina that are directly sensitive to light are the photoreceptor cells which are built of rods and cones. Light striking the retina triggers nerve impulses, which are sent to the brain through the optic nerve, the single route by which

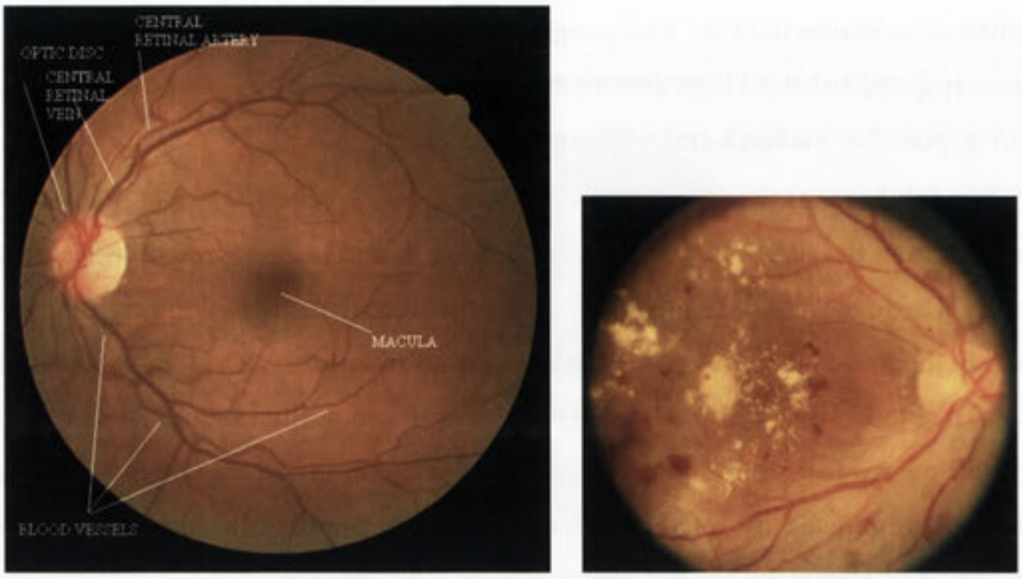
information leaves the eye. The ganglion cells transmit the image to the brain. The cones respond to bright light and mediate color vision. The rods respond to dim light and responsible for black and white night vision.

In adult humans the entire retina is 72% of a sphere about 22mm in diameter. The fovea defines the center of the retina (see Fig. A.1), and is the region of highest visual acuity. The fovea is directed towards whatever object you are looking right now. The ratio in the fovea between ganglion cells to photo-receptors is 2 to 1, the highest in the eye. The blood vessels also bypass a wide margin around the fovea. The area in and around the fovea is called macula. Most cones lie on the fovea whereas peripheral vision is dominated by rods. When using fundus camera, the fovea can be easily discerned as a centrally located area of darker pigmentation.

The ganglion cell axons all leave the eyeball at one location, the optic disk. The optic disc appears as an oval white area of $3mm^2$ and has no photo-receptors. As a result, it creates a hole in our vision, called the blind spot. Normally each eye compensates for the other, and the brain fills in the missing data, so we do not normally notice the blind spot. It is easy to make an experiment to see this blind spot by drawing two circles about 10cm apart on a piece of paper. Then we look at the left circle with the right eye and close the left eye. As we move our eyes closer to the paper, we notice that at one stage the right circle disappears. If we keep moving the paper closer the right circle will appear again. This is the blind spot.

A.1.2 Blood vessels

The retina is a multilayered membrane measuring from 0.10 to 0.23mm thick. It transforms light energy into chemical impulses that travel to the brain through the optic nerve. The retina is nourished by two sets of blood vessels: the retinal blood vessels and the choroidal blood vessels. The retinal blood vessels are those that are well evident through the whole of the image (see Fig. A.2). The central retinal artery enters the globe through the optic disc. It branches into the outbound retinal arteries (thinner, lighter blood vessels) and returning retinal veins (thicker, darker blood vessels). The choroid describes a network of spongy blood vessels that are separated from the



(a) (b)

Figure A.2: *Fundus images. (a) Image of a healthy retina. (b) Image of an unhealthy retina. This patient has advanced symptoms of diabetic retinopathy.*

retina by a layer of pigments cells.

The central retinal artery emerges from the center of the disc and immediately divides to deliver blood to the four quadrants of the retina. The superior and inferior temporal branches sweep around the fovea and roughly define the macula. The retinal veins form a similar pattern as they exit the eye at the optic disc. The optic disc can therefore always be found by tracing these vessels to their point of convergence.

Detection of blood vessels in retinal images offers a non-invasive method to diagnose various diseases in the retina. Segmenting the blood vessels can be used as the first step for subsequent image processing as they cover big areas of the retinal image and are the most stable structures appearing in fundus images. The leading cause of retina related blindness in the U.S. is due to Diabetic Retinopathy (DR) and age-related macular degeneration (AMD). It is believed that half of all blindness cases can be prevented by early detection. For example, the automatic detection of lesions such as microaneurysms, which are the first symptoms to appear in diabetic retinopathy can rely on vessel segmentation [52, 115]. Also, changes in the width of the vessels are an indication of ocular diseases such as glaucoma and diabetic retinopathy. Another

usage is for registering a number of retinal images from different perspectives and creating a 3D image of the retinal surface [104, 125]. The registered image can be used to measure the depression in the optic disc for the detection of glaucoma.

A.2 Previous work

Methods for segmenting blood vessels can be roughly divided into those that trace the center of the vessels [20, 24, 33, 95], methods based on learning and classification of feature vectors [101, 103, 117] and segmenting the vessel boundaries by using some set of filters or thresholds [19, 22, 50, 55, 87, 101]. Some of the proposed methods can span into more than one category. The tracking methods usually find a set of seed points and then use them to trace the retinal vasculature. The algorithm described in [24] is most closely related to our work, as both methods use the Kalman filter for tracking and matched Gaussian filters for detecting the blood vessels. In [24], the seed points are found in the circumference of the optic disc and the center of the vessel is tracked using an extended Kalman filter. The tracing is performed by using matched filters and detecting branching points. Tracing stops when the response from the Gaussian filter is low. However, our method differs in three important key points:

- We find seed points all over the image and do not rely on the hard task of tracing the vessel for the whole of its length.

- The next state in the tracking is based not only on the response from the matching filters, but also on the distribution of the gradients in a window around the vessel. It is helpful when tracking vessels that the interior is brighter than the background (unlike regular vessels).

- The tracking method is tolerant to vessels with areas of low response to the filters (for example, thin vessels with varying contrast) by ending the tracing only after a few consecutive bad responses. However, segmentations which have little resemblance to a vessel after a few steps are retraced.

A.3 Proposed method

An overview of our proposed method is as follows:

1. Seed point detection - Find a set of seed points by convolving the image with a group of matched filters in different widths and orientations. Unlike many other algorithms, we find as many as possible correct seed points, and do not limit them to certain areas (such as only the optic disc area). The aim is to have at least one seed point at every vessel, therefore removing the need to follow all branches.

2. Tracking the seeds - Track the center of the blood vessel by using the Kalman filter [118] starting from the set of seed points found in the previous step:

   (a) Estimate the next location using gradient information from both ends of the edges of the blood vessel.

   (b) Refine the estimation by correlating a cross-section of the estimated vessel with the shape of a vessel.

   (c) Estimate the vessel's width using the correlation results.

3. Stopping criterion - Stop the tracking if the likelihood of tracing a vessel is small for a number of consecutive steps or when we hit an already segmented vessel.

4. Retracing - Remove the segmented path originated from a seed point, if the tracking stopped in less than the minimum number of steps.

A.3.1 Detection of seed points

Matching filters have been used before [22] for the detection of blood vessels, and we use them as well, mainly for the detection of the seed points. A cross-section of a blood vessel will usually have the darkest value in the middle and gradually brighter values as approaching the edges (see Fig. A.3(a)). Outside the vessel we expect to have further increase in intensity. The cross-section can be thought as a function having a negative value inside the blood vessel and a positive value (or zero) outside the vessel.

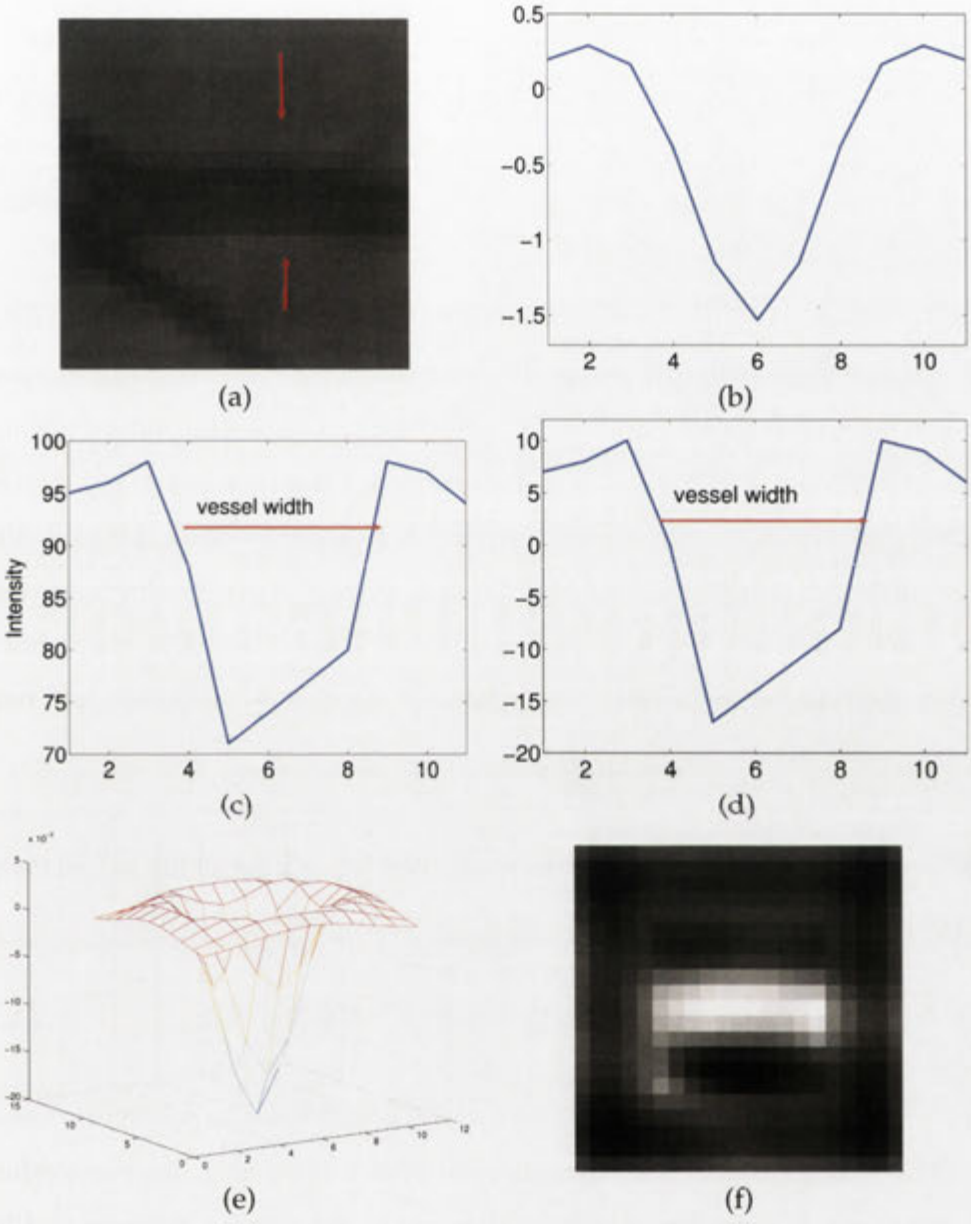


Figure A.3: *(a) An enlarged section of a retinal image showing a blood vessel. The arrows point to a cross-section of the vessel, which is 4 or 5 pixels wide. (b) A discrete Laplacian matching filter. The minimum occurs in the center and it has two zero-crossings. (c) The intensity of the cross-section of the blood vessel from (a). (d) Same as (c) after subtracting the average intensity value of the vessel. Most of the vessel's pixels have a negative value. (e) A 2D horizontal Laplacian filter. (f) Result of correlating the image (a) with the filter in (e). The brightest areas correspond to maximum correlation which happens in the center of the vessel.*

The aim of a matching filter is to find regions in the image that match the filter's shape. We therefore build the matching filters to have a similar shape to the blood vessel's cross-section. Fig. A.3(c) shows he pixel's intensity along a cross-section of the vessel from part (a). Fig. A.3(d) is obtained from Fig. A.3(c) after subtracting the average intensity of the cross-section from each pixel's intensity. It can be seen that the vessel has mostly negative values and the pixels outside the vessel has positive values.

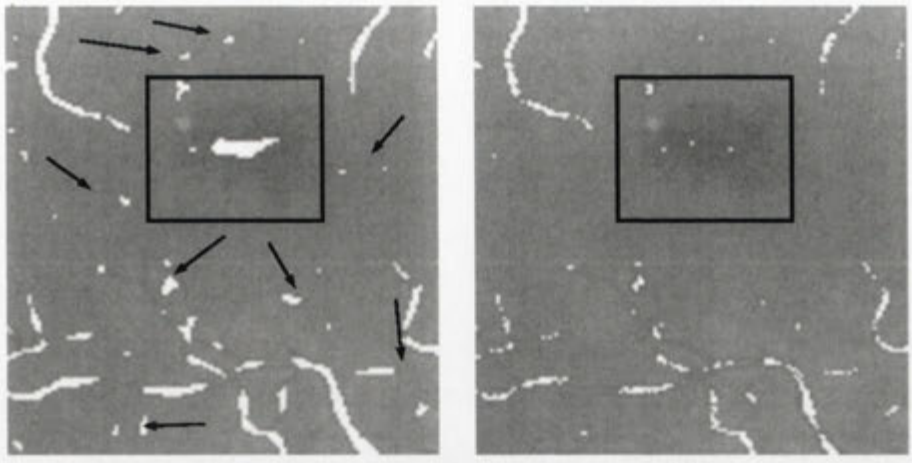
The filter's kernel is approximated by using a second order derivative Gaussian filter, $k(x;\sigma)$ as described in [24], where the value of σ is half of the estimated vessel width. Such kernel has its lowest value in the middle and increasing values towards the ends of the kernel. The zero-crossing points of the kernel correspond to the vessel's edges. The kernel is basically a Laplacian filter as can be seen in Fig. A.3(b). When convolving the kernel with the blood vessel, the response is the highest where the filter matches the vessel's length and orientation. This is due to the shape of the blood vessel and the kernel: both having negative values inside the vessel and positive values outside the vessel.

Because the vessels are considered to be piecewise linear segments, we create a 2D kernel by using a number of cross-sections. We define a set of 2D kernels with different orientations and widths:

$$k_{ij}(x, y; \theta_i, \sigma_j), \qquad 0 \leq \theta_i < 180 \qquad 2\sigma_j = 2, 4, 6, 8, \tag{A.1}$$

in order to cater for different vessels' widths and orientations. The width of a vessel can be as low as one pixel, and usually not wider than 9-10 pixels. We use 6 different orientations for θ spaced in 30 degrees from each other and 4 different widths, therefore we have in total 24 kernels.

The kernel's size is defined as 11×11 when $2\sigma_i < 8$ and 17×17 when $2\sigma_i = 8$, as we expect to have a longer continuous vessel structure at the same orientation for wide vessels and a larger σ requires a bigger kernel to represent it correctly. The kernels' size is based on measuring the minimum and maximum widths of the blood vessels in the given dataset. The kernel is normalized so the sum of its elements is 0



(a) (b)

Figure A.4: *The detected seed points around the macular area (in white): (b) after using (A.3) (c) after further using gradient information. The macula is framed by the black square and the arrows point to incorrect seeds. Most of the erroneous seeds are removed, and the rest should not be developed at the tracking step. Some legitimate seeds have been removed as well, but they are still surrounded by seed points. The final segmentation of this image appears in the third column of Fig. A.9.*

and variance is 1, therefore higher intensity pixels would not dominate the convolution response. We then convolve each of the 6 oriented filters with a given width with the image:

$$C(x, y; \theta_i) = k(x, y; \theta_i, \sigma) * f(x, y), \qquad 1 \leq i \leq 6 \tag{A.2}$$

For example, if a horizontal blood vessel with a width w is convolved with a horizontal 2D matching filter with $\sigma = w/2$ (see Fig. A.3(e)), the maximum response should occur in the center of the blood vessel. This can be seen in Fig. A.3(f) where the brightest pixels correspond to maximum correlation and are mostly in the vessel's center.

Only the responses that are above a threshold in a certain direction and have weaker response in the orthogonal direction are taken:

$$C(x, y; \theta_i) > T_s \text{ and } C(x, y; \theta_j) < T_s, \quad |\theta_i - \theta_j| = 90. \tag{A.3}$$

The threshold T_s should not be too high, because then no seed points are found in thin, low contrast vessels. It is generally preferable to use a mild threshold and then remove the false seed points during the tracking (see section A.3.3).

Each pixel found by a kernel $k_{ij}(x, y; \theta_i, \sigma_j)$ is expected to be in a center (or close to) of a blood vessel. To validate that, a cross-section in the direction normal to θ_i and of length $2\sigma_j$ is built. Therefore, the pixels at each end of the cross-section should correspond to the edges of the blood vessel. The angles of the gradients at these locations are compared and expected to be roughly in opposite directions. Finally, a small percentage of the brightest seed points is removed to avoid taking seeds that are between bright lesions or on the optic disc.

The process is repeated for the 4 different vessel widths given by σ. For each seed point we record the vessel's estimated width and orientation to be used for the tracking. It is possible for a seed point to have various widths or orientations, so only the one with the best response is taken.

Detecting seed points is usually quite fast and takes only about 10 percent of the running time. An example of the seed points that are found around the macular area (which does not contain vessels) after using (A.3) is shown in Fig. A.4(a). The remaining seed points in this area after applying further validation (gradient information) are shown in Fig. A.4(b). Most of the seeds in the macular region are removed and the rest of the erroneous pixels are expected to be removed during the tracking step. In Fig. A.5(a), a magnified region around the main vessel is shown with seed points appearing as vectors. The vector's orientation and length are related to the vessels orientation and length. Longer arrows correspond to longer vessels.

A.3.2 The Kalman Filter

Starting from a detected seed point with an estimated vessel width and orientation, the aim is to predict the trajectory of the center of the vessel. By considering the vessel tracking as a time series, where each state is a location in the vessel, it can be modelled by a state-space approach, such as the Kalman filter [118, 58]. The Kalman filter has been widely used to solve computer-vision related time series problems, such as road tracking, where the center of the road has to be detected [126]. Our method uses the traditional Kalman filter as the basis for the tracking, but we add additional error models that fit better to the problem in hand.

Figure A.5: *A zoomed area of the vasculature from Fig. A.9(a) middle row, showing the detected seed points (as vectors) and their orientation. The vector's length is related to the estimated vessel's width.*

At each step, the measurement is computed by examining the vessel structure (see Sec. A.3.3), and deciding the next location and vessel orientation by combining the prediction and the measurement. The evolution of the tracking process is defined by:

$$\mathbf{x_k} = a_k(\mathbf{x_{k-1}}) + \mathbf{w_{k-1}}, \tag{A.4}$$

where $\mathbf{x_k}$ is the state vector at time k (location and orientation in the blood vessel), $\mathbf{w_k}$ the system noise and $\mathbf{a_k}$ is a function of the state. Given a measurement $\mathbf{z_k}$ (next location in the vessel), the relationship between the state and the measurement is given by:

$$\mathbf{z_k} = h_k(\mathbf{x_k}) + \mathbf{v_k}, \tag{A.5}$$

where $\mathbf{v_k}$ is the measurement noise. The system noise and the measurement noise are assumed to be independent with covariance matrices Q_k and R_k. The system noise and the measurement noise can change throughout time, but are fixed in our implementation.

For a given seed point with estimated vessel width 2σ and orientation θ, the cur-

rent state $\mathbf{x_k}$ and the transition matrix are given by:

$$\mathbf{x_k} = [x, y, \sin(\alpha), \cos(\alpha)]^T, \qquad A = \begin{bmatrix} 1 & 0 & \Delta & 0 \\ 0 & 1 & 0 & \Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

so the prediction for the next state is $\mathbf{x_k} = A\mathbf{x_{k-1}}$ (without the noise). The current tracking direction is given by α, and Δ represents the step size at each iteration of the filter, and is initialized to $\Delta = \sigma$. Therefore the prediction for the next step is to move Δ pixels in the same direction as we moved before. The use of 4 elements in $\mathbf{x_k}$ (and not 3), is to be able to use the linear version of the Kalman filter. The measurement vector $\mathbf{z_k}$ which is the observation for the next vessel center, and H, the measurement model, are given by:

$$\mathbf{z_k} = (x, y)^T, \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

The matrix H relates the state $\mathbf{x_k}$ to the measurement $\mathbf{z_k}$ via (A.5). We use a simple model with $H\mathbf{x_k}$ just returning the location in the blood vessel given by the state $\mathbf{x_k}$.

The system noise covariance matrix Q, is dependent on the vessel width: thinner vessels are assumed to have more noise in the moving direction, because their tendency to be more windy when compared to the wide vessels. The measurement noise covariance matrix R is set to 1 pixel in each direction. Both matrixes are fixed during the tracking and are given as follows:

$$Q = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & e_w & 0 \\ 0 & 0 & 0 & e_w \end{bmatrix}$$

$$e_w = \begin{cases} 0.6 & \text{if } 2\sigma \leq 4\text{pixels} \\ 0.2 & \text{otherwise} \end{cases}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Ending condition

At each step the measurement returns an error value, which is 0 if the tracking still follows a vessel and 1 otherwise (how the error value is computed is explained in next section). The tracing stops only after 3 consecutive errors, therefore letting the tracing continue in locations of uncertainty such as low gradient information, branching or fast change in direction. This empirical value proved to be suitable for our application, because by continuing an additional two tracing steps, the problematic section can be passed.

Finally, we define a value of the minimum number of steps that have to be completed by the tracker, in order to keep the segmentation. The aim is to remove segmentations from seed points which are not within a vessel at all and after a few moves no vessel texture is found. Such situation happens when the seed points are on the macula, on the optic disc or between two close vessels that can create a misleading shape of inner vessel. We define this value to equal to 9 steps, which is three times the number of accepted errors before terminating. We found out that this value will remove most of the erroneous vessels, but sometimes can also retrace short sections of valid vessels. When a decision to retrace the segmentation from a seed point is made, the whole segment is removed. This is equivalent to treating the seed point as a false one (or not detecting it at first place). Although the idea deviates from the standard Kalman filter definition, it suits our problem, since only after starting the tracking these seed points are detected as faulty.

A.3.3 Vessel tracking

In order to predict the next state $\mathbf{z}_k$, the Kalman filter calls a measurement function that estimates the next location, given the current location and orientation of tracking ($\mathbf{x}_k$). The function performs a number of correctness checks and if any of them fails, an error status is returned to the tracker as described in the previous paragraph.

Gradient computation

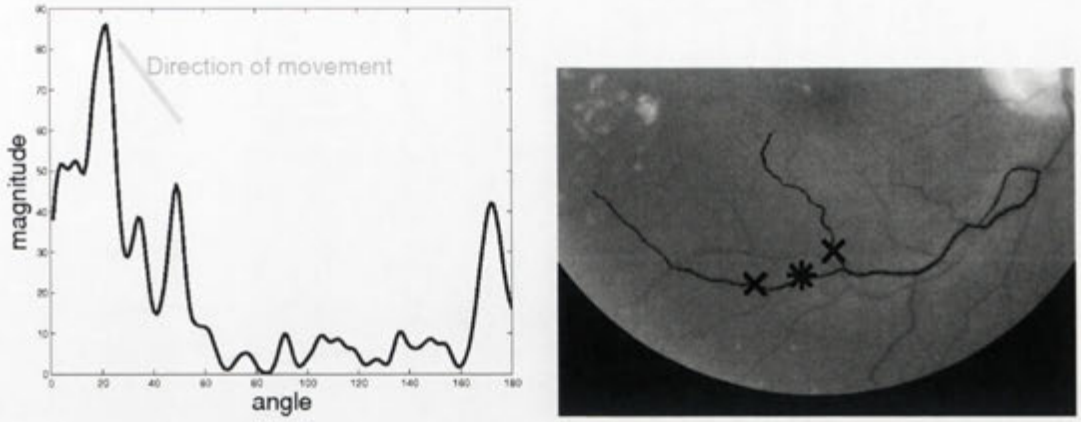
A window is centered about the current location $\mathbf{x}_k$ in the blood vessel. The window is of size $(2w+1) \times (2w+1)$, assuming the vessel's estimated width is w. In such a window, we except to have a small section of the blood vessel, which follows a certain direction. Thus, when examining one end of the vessel, we expect to have several gradients pointing towards the same orientation. Equally, we expect to have gradients pointing to the opposite direction at the other end.

The gradients G_x,G_y in the x and y directions respectively are computed in the window. We are interested in finding the orientation which has the strongest magnitude response and assume the vessel orientation is perpendicular to it. Therefore, the gradient's angle and magnitude are given by:

$$\theta = \arctan(G_y/G_x), \qquad \text{mag}(x,y) = \sqrt{G_x^2 + G_y^2} \tag{A.6}$$

However, in order to simultaneously take advantage of the gradient information from both ends of the vessel, we compute the double gradient angle, 2θ, for each pixel. The reason is that gradients of opposite ends of the vessel should differ by 180 degrees, so doubling the angle will make them equal and reinforce each other. If $\theta_1 = \theta_2 + 180$, then $2\theta_1 = 2\theta_2 + 360$, thus $2\theta_1 = 2\theta_2$, so the double angle is the same. Since the magnitude of the double angle equals the magnitude of the original angle, the gradient of the double angle (G_{2x} and G_{2y}) is related to the original gradient using simple trigonometry:

$$\begin{aligned} \frac{G_{2x}}{\sqrt{G_x^2 + G_y^2}} &= \cos(2\theta) = \cos^2\theta - \sin^2\theta = \frac{G_x^2 - G_y^2}{G_x^2 + G_y^2} \\ \frac{G_{2y}}{\sqrt{G_x^2 + G_y^2}} &= sin(2\theta) = 2\sin(\theta)\cos(\theta) = \frac{2G_x^2 G_y^2}{G_x^2 + G_y^2} \end{aligned} \tag{A.7}$$



(a) (b)

Figure A.6: *(a) Histogram of the magnitudes of the double angle for the pixel marked with a star in part (b). In this histogram the double angle is divided by two, therefore retrieving the original angle from 0 to 180. The highest peak is the estimated vessel's orientation. (b) Example of tracking the vessels using two seed points marked by X. The left seed has an initial width estimation of 7 pixels and the right one of 3 pixels. Only the center of the vessel has being segmented, so it is possible to see how the segmentations follow the center of the vessels and merge into one line and then separate again. Also the hollow vessel on the upper right hand side has been traced.*

Histogram computation

In order to find the best orientation for the next state, we use a similar method to the Parzen's window method. Our random variable is the double angle ranging from 0 to 360 degrees. The sample points are the double angles computed inside the window. For each angle we sum the corresponding pixel's magnitude as given by (A.6) (and not just the number of occurrences of the angle). Therefore, the histogram depicts the distribution of the double angle in the window, where each pixel contributes its magnitude. The histogram is smoothed by convolving with a Gaussian filter with a small variance. As we are looking at a section of a vessel we expect the histogram to have only one peak with a probability distribution exceeding a threshold, as most vessels have stronger gradient information than the surroundings. Fig. A.6(a) demonstrates such histogram for a window, where there is one dominant peak around 22 degrees. Thus, the original gradient angle is about 11 degrees, so the tracking direction is perpendicular to this angle.

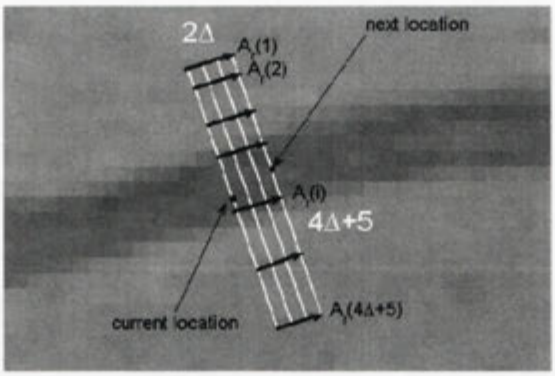


Figure A.7: *A blood vessel and its cross-section painted by the white lines. The black lines show the direction the pixels are averaged to create the vector A_r as explained in Sec. A.3.3. The next state found by the tracker after the validation step is pointed out.*

Validation by cross-section

The angle θ_m of the strongest peak in the histogram is taken. This angle is perpendicular to the vessel's orientation and is used to build a cross-section of the blood vessel at the current location x_k to validate and refine our estimation for the orientation. We use a similar idea to the one used to built the matching filters: A single cross-section is built as a vector of intensities of length $4\Delta + 5$ and orientation θ_m. The length of the cross-section is chosen so it will always span the whole width of the vessel (2Δ), even when the current state is located near the vessel's end. Utilizing the piecewise linearity of the vessel, and looking at a set of 2Δ parallel cross-sections, a matrix of size $2\Delta \times (4\Delta + 5)$ is created. For each column the average pixel value $A_r(x)$ is calculated, as shown in Fig. A.7 by the direction of the black arrows.

The vector $\mathbf{A}_r$ should be roughly shaped as an upside-down Gaussian, as the lowest intensities are excepted to be in the center. However, if the location of the next state is not estimated correctly using the histogram computation, the next state might be positioned towards one of the ends of the vessel. In that case, the Gaussian is not centered in the vector. As we are interested in the relative intensity difference between the pixels in the interior of the vessel and the pixels outside the vessel, we reverse the shape of the vector:

$$A_p(x) = max(\mathbf{A}_r) - A_r(x), \quad 1 \leq x \leq 4\Delta + 5 \tag{A.8}$$

This creates a Gaussian-shaped vector, where the highest values should correspond to the vessel's interior. In order to find the accurate center of the vessel, the vector $\mathbf{A}_p$ is convolved with a Gaussian with a σ that is related to the estimated width of the vessel. If the maximum value of the convolution is above a threshold, then the new tracking orientation is taken as pointing towards the point of maximum correlation (shown in Fig. A.7 as the next location). Otherwise, the orientation stays the same as the previous one. An example of how the center of the vessel is traced from two seed points until the ending condition is fulfilled is shown in Fig. A.6(b). The tracing did manage to continue tracing through some of the branching points and mostly to stick to the center of the vessels. In the left side the tracing stopped in a branching point probably due to weak gradient information.

Estimating the vessel's width

The vector $\mathbf{A}_p$ is scanned towards both ends, starting from the pixel with the highest correlation. This pixel should correspond to the center of the vessel. The aim is to define cut-off points at both sides of the vector which define the width of the vessel. A pixel is segmented only if:

$$A_p(x) > \min\{\max(\mathbf{A}_p)T_3, T_4\}. \tag{A.9}$$

The term $\max(\mathbf{A}_p)T_3$ defines the minimum intensity that should still be defined as part of the vessel (remember that the vector $\mathbf{A}_p$ is a Gaussian-shaped vector, where the highest intensities have the best probability of being a vessel). The term T_4 defines a threshold on the intensity difference from the background. Once a pixel does not comply with (A.9) the search in that direction is stopped. The thresholds T_3 and T_4 have strong influence on the result as they define the sensitivity to low contrast vessels and to pixels on the vessel's boundary. We found out that different thresholds give better results with different manual segmentations. We also incorporate our initial estimation of the vessel's width to control the number of pixels segmented.

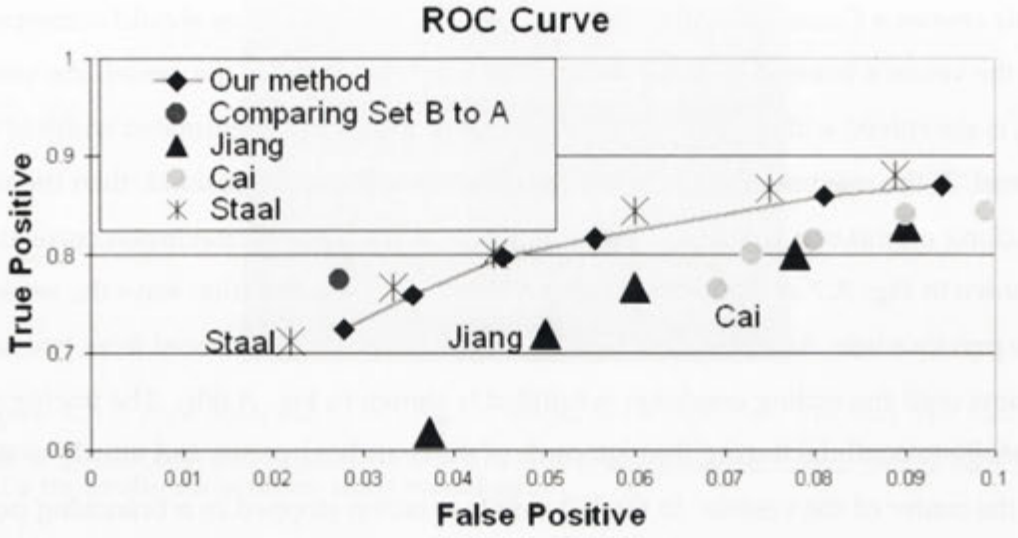


Figure A.8: *ROC curve of our algorithm. Our results are drawn by the continuous line. Other methods are plotted with unconnected points.*

A.4 Results

We tested our method on the public database of DRIVE [103], which includes 40 images: 20 training images and 20 test images. For the test images, 2 manual segmentations are given. Previously published results use the 20 testing images and compare their results to the first manual segmentation. We used the same images and manual segmentations and also the provided mask image for the retinal area to calculate the false positives (FP). Fig. A.8 shows the ROC curve for our algorithm. In order to induce the graph the parameters T_s, T_3 and T_4 were set and changed in (A.3) and (A.9). Their values were tested by running experiments on all images and estimating the range which provides valid results. As the manual segmentations are subjective, and there is only a 78% agreement between the two given manual segmentations, it makes sense to have parameters that will be finalized according to the given database. The threshold in (A.3) determines the number of seed points that are detected and whether low contrast vessels have seed points in them. The thresholds in (A.9) set the sensitivity when segmenting the vessel's width and mainly affect the number of boundary pixels that are segmented.

In the ROC curve we only show the part where the false positives are lower than 10 percent, as for high FP values the results become almost useless, since the segmented width of the vessels is very inaccurate and large areas which do not belong to vessels are segmented. This is especially true in the case of retinal images as the number of non-vessels pixels is much higher than the number of vessel pixels (up to 8:1). We managed to produce almost 80% of true positives (TP) with 4.5% of false negatives and 85.9% with 8.1% using a different set of parameters. Results of four images with their manual segmentations are shown in Fig. A.9. Most of the main and some small vessels are segmented and there are almost no segmented lesions or other retinal structures. The rightmost image in the figure is probably one of the most challenging images in the dataset, as it contains lesions and hollow vessels, which are hard to segment, since they are not darker than the background as normal vessels are. The last row shows the TP, FP and False Negatives (FN) in different colors over a white background. Most of the FP (green) are around actual vessels, therefore the vessels have been segmented, but do not share the exact border with the manual segmentation. Only small regions of green (FP) are retinal structures which are not vessels. The main difficulty is in segmenting the very thin vessels (red pixels). Some of these vessels might have seed points in them, but they do not develop due to poor gradient information. Also not all of these vessels appear in the second manual segmentation. It takes between 20sec to 40sec to segment an image, depending on the threshold used in (A.3) and the number of seed points found for the specific image.

We compare our results with recent publications that use the DRIVE database. All the results from the competing approaches were copied or estimated from the curves provided in the original publications, and are drawn as unconnected points in the ROC graph. Cai [19] computed the distance between the ROC curve to the ideal point (0,1) and compared their results to Jiang [55] and Staal [103]. Since their graph is plotted for FP < 0.5, it is not necessary that the best distance is found for small value of false positives. However, for the area where FP < 0.1 our results are superior to both the methods of Jiang and Cai, as can be seen in Fig. A.8. The results presented by Staal [103] are slightly superior to ours. However, their method is based on machine learning and requires a lengthy training time for accumulating data and

hand segmenting the images by clinicians. Furthermore, retraining may be necessary if the imaging method or technology changes. This is not needed by our method. We also compared our results with the second manual segmentation and found out that with all sets of parameters the TP rate was 2-2.5% higher with almost the same rate of FP. This result agrees with the subjectiveness of the manual segmentation.

Finally, I would like to briefly address the formula that has being used widely to measure the success of the segmentation through the ROC curve. By comparing the number of pixels that are segmented in both the manual and automatic segmentation, the TP rate is being biased and tends to produce high rates even if the segmentation is lacking. Since large areas of the images are certainly not vessels and the wide vessels are easier to segment and count for high percentage of the vessel's pixels, a relatively high TP rate can be achieved by just segmenting these vessels resulting in a low FP rate. However, the main issue is evaluating these results in terms of the actual usefulness of the segmentation. For example, if an application does not segment the thin vessels at all, it can be advantageous. As these vessels count only for a small percentage from the total number of vessel's pixels, the TP rate is not reduced dramatically by not segmenting them, but the FP rate is guaranteed not to increase. Furthermore, segmenting these thin vessels and being wrong by only a single pixel is counted as erroneous segmentation, thus increasing the FP rate and not the TP rate.

A different method to compare the success of the segmentation can count the total length of the vessels that has been segmented in the manual and automatic images. Another option is to follow the ideas presented in [101] where only the center of the vessel is extracted. Both methods are not biased by the large number of pixels in wide vessels, while thin vessels have to be segmented to achieve high TP rate.

A.5 Summary and conclusion

In this chapter we have presented a method for segmenting the blood vessels in retinal images. The vessels are detected using a tracking algorithm, which is implemented using the Kalman filter. The tracking is carried out from multiple starting points. Each seed point is found by using matching filters and gradient information and the

points are usually spread out over all the image. Some of the seeds might not be inside vessels, but are detected due to structures in the image that resemble a vessel. To that end, we introduce the idea of retracing seeds that their first few time-series states seem to be outside a vessel. Seed points that progress over the initial retracing cut-off point, contribute to the final segmentation. Tracking these seeds stops if we hit an already segmented vessel or when encountering a weak vessel's response for a few consecutive states. Not terminating immediately tracings that have a weak response overcomes local weak gradient information, illumination changes and regions of branching. Finally, we show that our algorithm provides good segmentation results when compared to other methods that used the DRIVE database. We manage to segment thin vessels and to avoid segmenting other retinal structures.

Our method can be improved by introducing a better algorithm to estimate the correct width of the blood vessel. Our current method uses thresholds, that do not consider the smoothness of the segmentation or knowledge regarding transitions from wide to thin vessels. A graph-cut based algorithm for segmenting the vessel's width can use a unary term which is based on (A.8) and (A.9). The pairwise term can enforce smoothness, thus changes from a thick to a thin width will not happen immediately and the vessel will have a gradual change of width. In addition, provided we have some knowledge about the image, a monotonic constraint can be introduced, which will enforce non-increasing vessel's width. Thus, we assume that vessels usually have a constant width and the transition in branching areas is only to vessels with a smaller width.

Finally, the method should be tested on other databases. We believe that the presented algorithm will work well on other databases, however adjustments to some of the thresholds are needed.

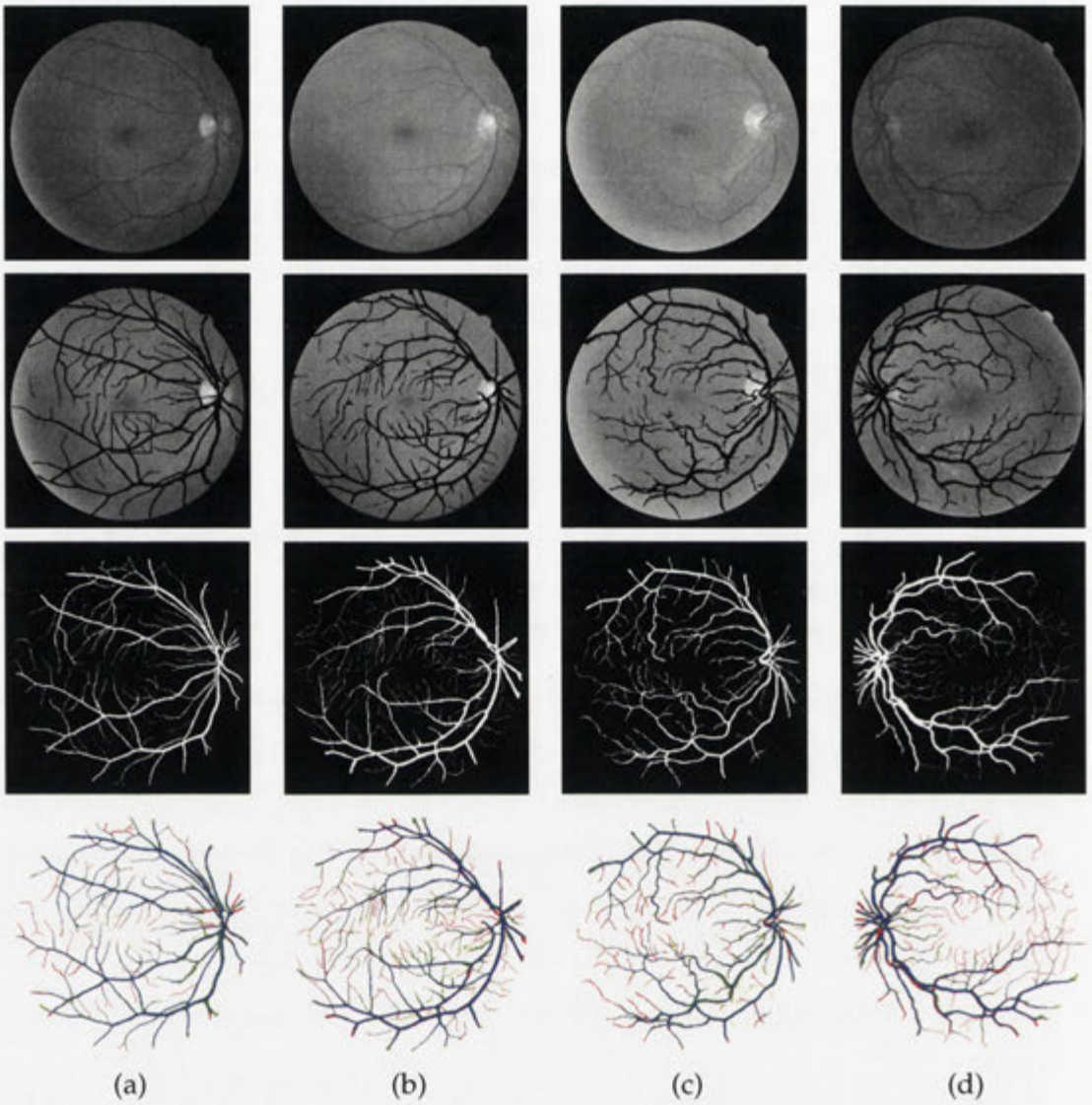


Figure A.9: *Segmentation results.* ***First row:*** *original images.* ***Second row:*** *Segmentation results for 80% TP and 4.5% FP. Our segmentation appears as black pixels on top of the original image.* ***Third row:*** *The manual segmentation provided by the first clinician (gold standard).* ***Fourth row:*** *TP are shown in blue, FN in red and FP in green. In columns (a) and (b) the images are of healthy retinas. In column (c) the image has hollow vessels, which are hard to segment. In column (d) the image contains lesions, hollow vessels and many thin vessels. In the last column, most of the green pixels appear next to the manually segmented vessels and only few are of other structures. The red pixels are mainly very thin vessels, which are hard to segment with a low FP rate. These vessels probably require a different segmentation approach, which is more sensitive.*

Bibliography

[1] M. Abelson, G. Ousler, and L. A. Nally. Alternate reference values for tear film break-up time in normal and dry eye populations. *Adv Exp Med Biol*, 506:1121–1125, 2002.

[2] A. Abhyankar and S. Schuckers. Active shape models for effective iris segmentation. *Proc. of SPIE, Biometric technology for human identification III*, 6202, 2006.

[3] A. M. Ali, A. A. Farag, and A. El-Baz. Graph cuts framework for kidney segmentation with prior shape constraints. In *MICCAI (1)*, pages 384–392, 2007.

[4] E. M. Arvacheh and H. R. Tizhoosh. Iris segmentation: Detecting pupil, limbus and eyelids. *Proc. Int. Conf. on Image Processing*, pages 2453–2456, 2006.

[5] S. Baker and I. Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221 – 255, March 2004.

[6] C. G. Begley, B. Caffery, K. K. Nichol, and R. Chalmers. Responses of contact lens wearers to a dry eye survey. *Optom. Vis. Sci.*, 77:40–46, 2000.

[7] C. G. Begley, N. Himebaugh, D. Renner, H. Liu, R. Chalmers, T. Simpson, and J. Varikooty. Tear breakup dynamics: a technique for quantifying tear film instability. *Optom Vis Sci*, 83(1):15–21, 2006.

[8] V. Bijsterveld. Diagnostic tests in the sicca syndrome. *Arch Ophthalmol*, 82:10–14, 1969.

[9] E. Bitton, A. Keech, T. Simpson, and L. Jones. Variability of the analysis of the tear meniscus height by optical coherence tomography. *Optom Vis Sci.*, 84(9):903–908, 2007.

[10] E. Bitton and V. Lovasik. Longitudinal analysis of precorneal tear film rupture patterns. *Adv Exp Med Biol.*, 438:381–389, 1998.

[11] C. S. Blehm, S. Vishnu, and A. Khattak. Computer vision syndrome: a review. *Surv Ophthalmol.*, 50:253–262, 2005.

[12] M. Bleyer and M. Gelautz. Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions. *Signal Processing: Image Communication*, 22:127–143, 2007.

[13] E. Boros and P. L. Hammer. Pseudo-boolean optimization. *Discrete Appl. Math.*, 123(1-3):155–225, 2002.

[14] C. A. Bouman. Markov random fields and stochastic image models. *IEEE ICIP*, 1995.

[15] Y. Boykov and G. Funka-Lea. Graph cuts and efficient n-d image segmentation. *Int. J. Comput. Vision*, 70(2):109–131, 2006.

[16] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *ICCV*, pages 105–112, 2001.

[17] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on PAMI.*, 23(11):1222–1239, 2001.

[18] A. J. Bron, M. B. Abelson, E. Pearce, G. Ousler, A. Tomlinson, N. Yokoi, J. A. Smith, C. Begley, B. Caffery, K. Nichols, D. Schaumberg, and O. Schein. Methodologies to diagnose and monitor dry eye disease: Report of the diagnostic methodology subcommittee of the international dry eye workshop (2007). *The Ocular Surface*, 5(2):108–152, 2007.

[19] W. Cai and A. Chung. Multi-resolution vessel segmentation using normalized cuts in retinal images. *MICCAI*, pages 928–935, 2006.

[20] A. Can, H. Shen, J. N. Taylor, H. L. Tanenbaum, and Roysam. Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithms. *IEEE Trans. on Info. Tech. in Biomedicine*, 3(2):125–138, 1999.

[21] P. Carr and R. Hartley. Improved single image dehazing using geometry. In *DICTA09*, 2009.

[22] S. Chaudhuri, S. Chatterjee, N. Katz, N. Nelson, and M. Goldbaum. Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Trans. on Med. Imag.*, 8(3):263–269, 1989.

[23] E. M. Chia, P. Mitchell, and E. Rochtchina. Prevalence and associations of dry eye syndrome in an older population: the blue mountains eye study. *Clin Experiment Ophthalmol.*, 31:229–232, 2003.

[24] O. Chutatape, L. Zheng, and S. Krishnan. Retinal blood vessel detection and tracking by matched gaussian and kalman filters. *IEEE Eng. in Med. and Bio. Soc.*, 20:3144–3149, 1998.

[25] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam. The use of active shape models for locating structures in medical images. *Information Processing in Medical Imaging (IPMI)*, pages 33–47, 1993.

[26] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms (Second ed.)*. MIT Press and McGraw-Hill, 2001.

[27] J. Daugman. The importance of being random: statistical principles of iris recognition. *Pattern Recognition*, 36(2):279–291, 2003.

[28] J. Daugman. New methods in iris recognition. *IEEE Trans. on Systems, Man and Cybernetics - Part B: Cybernetics*, 37(5), 2007.

[29] J. G. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. on PAMI*, 15(11):1148–1161, 1993.

[30] M. Di Pascuale, E. M. Espana, and S. C. G. Tseng. Clinical characteristics of conjunctivochalasis with or without aqueous tear deficiency. *Br. J. Ophthalmol.*, 88:388–392, 2004.

[31] M. M. Fahim, S. Haji, C. V. Koonapareddy, V. C. Fan, and P. A. Asbell. Fluorophotometry as a diagnostic tool for the evaluation of dry eye disease. *BMC Ophthalmology*, 6:1–7, May 2006.

[32] X. Feng, C. Fang, X. Ding, and Y. Wu. Iris localization with dual coarse-to-fine strategy. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 553–556. IEEE Computer Society, 2006.

[33] M. Franz and R. Schuffny. Segmentation of blood vessels in subtraction angiographic images. *DICTA*, pages 215–224, 2003.

[34] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, November 1984.

[35] E. Goto and S. C. G. Tseng. Kinetic analysis of tear interference images in aqueous tear deficiency dry eye before and after punctal occlusion. *IOVS*, 44(5):1897–1905, 2003.

[36] L. Grady and M. P. Jolly. Weights and topology: A study of the effects of graph construction on 3d image segmentation. *MICCAI*, pages 110–118, 2008.

[37] J. P. Guillon. Tear film structure of the contact lens wearer. *PhD Thesis, The City University London*, 1987.

[38] J. P. Guillon. Non-invasive tearscope plus routine for contact lens fitting. *Contact Lens and Aterior Eye*, 21:31–40, 1998.

[39] J. P. Guillon and G. Young. Subtle signs of sicca – advanced tear film assessment. *Contact Lens Spectrum*, 1999.

[40] C. Harris and M. Stephens. A combined corner and edge detection. In *Proceedings of The Fourth Alvey Vision Conference*, pages 147–151, 1988.

[41] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[42] Z. He, T. Tan, and Z. Sun. Iris localization via pulling and pushing. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 366–369. IEEE Computer Society, 2006.

[43] Z. He, T. Tan, Z. Sun, and X. Qiu. Robust eyelid, eyelash and shadowlocalization for iris recognition. *Proc. Intl. Conf. Image Processing (ICIP)*, pages 265–268, 2008.

[44] Z. He, T. Tan, Z. Sun, and X. Qiu. Toward accurate and fast iris segmentation for iris biometrics. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 31(9):1670–1684, 2009.

[45] N. L. Himebaugh, A. R. Wright, A. Bradley, C. G. Begley, and L. N. Thibos. Use of retroillumination to visualize optical aberrations caused by tear film breakup. *Optom. Vis. Sci.*, 80(1):69–78, 2003.

[46] F. J. Holly. Formation and rupture of the tear film. *Exp. Eye Res.*, pages 515–525, 1973.

[47] F. J. Holly. Surface chemical evaluation of artificial tears and their ingredients. i. interfacial activity. *Con. and Intraoc. Lens Med. Journal*, 4(2):14–31, 1978.

[48] F. J. Holly. Tear film physiology and contact lens wear. pertinent aspects of tear film physiology. *Am J Optom & Physiol Opt.*, 58(4):324–330, 1981.

[49] F. J. Holly and M. A. Lemp. Tear physiology and dry eyes. *Surv. Ophthal.*, 22:69–87, 1977.

[50] A. Hoover, V. Kouznetsova, and M. Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Trans. on Med. Imag.*, 19(3):203–210, 2000.

[51] J. Huang, Y. Wang, T. Tan, and J. Cui. A new iris segmentation method for recognition. *Proc. Intl. Conf. on Pattern Recogntion (ICPR)*, pages 554–557, 2004.

[52] K. Huang and M. Yan. A local adaptive algorithm for microaneurysms detection in digital fundus images. In *CVBIA*, pages 103–113, 2005.

[53] H. Ishikawa. Exact optimization for markov random fields with convex priors. *PAMI*, 25(10):1333–1336, 2003.

[54] D. R. Iskander, M. J. Collins, and B. Davis. Evaluating tear film stability in the human eye with high-speed videokeratoscopy. *IEEE Transactions on Biomedical Engineering*, 52(11):1939–1949, 2005.

[55] X. Jiang and D. Mojon. Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images. *PAMI*, 25:131–133, 2003.

[56] M. P. Jolly and L. Grady. 3d general lesion segmentation in ct. *IBSI*, pages 796–799, 2008.

[57] A. Jordan and J. Baum. Basic tear flow: does it exist? *Opthal.*, 87:920–930, 1980.

[58] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, pages 182–193, 1997.

[59] L. Kang, W. Xiaodong, D. Z. Chen, and M. Sonka. Optimal surface segmentation in volumetric images: A graph-theoretic approach. *IEEE Trans. on PAMI*, 28(1), 2006.

[60] P. Kohli, L. Ladicky, and P. H. S. Torr. Robust higher order potentials for enforcing label consistency. *CVPR*, 2008.

[61] P. Kohli, J. Rihan, M. Bray, and P. H. Torr. Simultaneous segmentation and pose estimation of humans using dynamic graph cuts. *Int. J. Comput. Vision*, 79(3):285–298, 2008.

[62] P. Kohli and P. H. S. Torr. Efficiently solving dynamic markov random fields using graph cuts. *ICCV*, pages 922–929, 2005.

[63] V. Kolmogorov and Z. Ramin. Computing visual correspondence with occlusions using graph cuts. *ICCV*, 2001.

[64] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. *ECCV*, pages 82–96, 2002.

[65] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Trans. on PAMI*, 26(2):147–159, 2004.

[66] V. Kolmogorov and R. Zabih. Graph cut algorithms for binocular stereo with occlusions. *Mathematical Models in Computer Vision: The Handbook*, pages 423–438, 2005.

[67] W. K. Kong and D. Zhand. Accurate iris segmentation based on novel reflection and eyelash detection model. *Proc. of Intl. Symposium on Intelligent Multimedia, Video and Speech Processing*, pages 263–266, 2001.

[68] D. R. Korb, J. Craig, M. Dought, J. P. Guillon, G. Smith, and A. Tomlinson. *The Tear Film*. Butterworth Heinemann, first edition, 2002.

[69] D. R. Korb, J. V. Greiner, and J. Herman. Comparison of fluorescein break-up time measurement reproducibility using standard fluorescein strips versus the dry eye test (DET) method. *Cornea*, 20(8):811–815, 2001.

[70] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Obj cut. In *CVPR 2005*, pages 18–25, Washington, DC, USA, 2005. IEEE Computer Society.

[71] J. D. Lafferty, A. McCallum, and F. C. N. Periera. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*, pages 282–289, 2001.

[72] D. W. Lamberts, C. S. Foster, and P. H. D. Schirmer test after topical anesthesia and the tear meniscus height in normal eyes. *Arch Ophthalmol.*, 97(6):1082–1085, 1979.

[73] M. A. Lemp. Report of the national eye institute/industry workshop on clinical trials in dry eyes. *CLAO J*, 21:221–232, 1995.

[74] M. A. Lemp, C. H. Dohlman, and F. J. Holly. Corneal desiccation despite normal tear volume. *Ann Ophthalmol*, 2:258–261, 1970.

[75] M. A. Lemp and J. R. Hamill. Factors affecting tear film break up in normal eyes. *Arch Ophthalmol*, 89(2):103–105, 1973.

[76] S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer-Verlag, 2001.

[77] M. H. Lin and C. Tomasi. Surfaces with occlusions from layered stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):1073–1078, 2004.

[78] X. Liu, K. W. Bowyer, and P. J. Flynn. Experiments with an improved iris segmentation algorithm. In *AUTOID '05: Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, pages 118–123, Washington, DC, USA, 2005. IEEE Computer Society.

[79] X. Liu, O. Veksler, and J. Samarabandu. Graph cut with ordering constraints on labels and its applications. *CVPR*, 2008.

[80] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[81] G. Loy and A. Zelinsky. A fast radial symmetry transform for detecting points of interest. In *ECCV*, pages 358–368, London, UK, 2002. Springer-Verlag.

[82] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proceedings of Imaging understanding workshop*, pages 121–130, 1981.

[83] L. Ma, T. Tan, Y. Wang, and D. Zhang. Efficient iris recognition by characterizing key local variations. *IEEE Transactions on Image Processing*, 13(6):739–750, 2004.

[84] J. C. Mainstone, A. S. Bruce, and T. R. Golding. Tear meniscus measurement in the diagnosis of dry eye. *Current Eye Research*, 15(6):653–661, 1996.

[85] R. Marquardt, R. Stodtmeiser, and T. Christ. Modification of tear film break-up time test for increased reliability. *The Preocular Tear Film in Health, Disease and Contact Lens Wear*, pages 57–63, 1986.

[86] J. E. McDonald and S. Brubaker. Meniscus induced thinning of tear films. *Am. J. Opth.*, 72(1):139–146, 1971.

[87] A. N. Mendonca and A. Campilho. Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction. *IEEE. Trans. on Med. Imag*, 25(9):1200–1213, 2006.

[88] B. Miljanovic, K. A. Trivedi, and M. R. Dana. Relation between dietary n-3 and n-6 fatty acids and clinically diagnosed dry eye syndrome in women. *Am. J. Clin. Nutr.*, 82:887–893, 2005.

[89] D. Miller and S. C. G. Tseng. Conjunctivochalasis: Literature review and possible pathophysiology. *Surv Ophthalmol*, 43(3):225–232, 1998.

[90] H. Milton. Hallway controversies in dry eyes. *Review of Optometry*, 2003.

[91] M. Norn. Desiccation of the pre corneal film. *Acta Ophthal.*, 47(4):865–880, 1969.

[92] J. R. Palakuru, J. Wang, and J. V. Aquavella. Effect of blinking on tear dynamics. *Invest Ophthalmol Vis Sci*, 48(7):3032–3037, 2007.

[93] S. C. Pflugfelder, G. Geerling, S. Kinoshita, M. A. Lemp, J. McCulley, D. Nelson, G. N. Novack, J. Shimazaki, and C. Wilson. Management and therapy of dry eye disease: Report of the management and therapy subcommittee of the international dry eye workshop (2007). *The Ocular Surface*, 5(2):161–176, 2007.

[94] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007.

[95] P. Quelhas and J. Boyce. Vessel Segmentation and Branching Detection using an Adaptive Profile Kalman Filter in Retinal Blood Vessel Structure Analysis. In *IbPRIA 2003, LNCS*, volume 2652, pages 802 – 809, 2003. IDIAP-RR 03-58.

[96] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[97] C. Rother, V. Kolmogorov, and A. Blake. Grabcut - interactive foreground extraction using iterated graph cuts. *ACM Trans. on Graphics*, 23(3):309–314, 2004.

[98] M. Shen, J. Wang, A. Tao, Q. Chen, S. Lin, J. Qu, and F. Lu. Diurnal variation of upper and lower tear menisci. *Am J Ophthalmol*, 145(5):801–806, 2008.

[99] D. Singaraju, L. Grady, and R. Vidal. Interactive image segmentation via minimization of quadratic energies on directed graphs. *ECCV*, 2008.

[100] J. A. Smith, J. Albeitz, C. Begley, B. Caffery, K. Nichols, D. Schaumberg, and O. Schein. The epidemiology of dry eye disease: Report of the epidemiology subcommittee of the international dry eye workshop (2007). *The Ocular Surface*, 5(2):93–107, 2007.

[101] M. Sofka and C. V. Stewart. Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures. *IEEE Trans. on Med. Imag.*, 25(12):1531–1546, 2006.

[102] A. Sommer. Vitamin a deficiency and the global response. *Forum Nutr.*, 56:33–35, 2003.

[103] J. Staal, M. D. Abramoff, M. Niemeijer, M. A. Viergever, and B. van Ginneken. Ridge based vessel segmentation in color images of the retina. *IEEE Trans. on Med. Imag.*, 23(4):501–509, 2004.

[104] C. V. Stewart, C. Tsai, and B. Roysam. The dual-bootstrap iterative closest point algorithm with application to retinal image registration. *IEEE Trans. on Med. Imag*, 22:2003, 2003.

[105] R. L. Terry, C. M. Schnider, and B. A. Holden. The CCLRU standards. *Optician*, 206(5430):18–23, 1993.

[106] J. M. Tiffany and A. J. Bron. Role of tears in maintaining corneal integrity. *Tans. Opthal. Soc. U.K.*, 98:335–338, 1978.

[107] C. L. Tisse, L. Martin, L. Torres, and M. Robert. Person identification technique using human iris recognition. In *Proc. of Vision Interface*, pages 294–299, 2002.

[108] E. Trucco and M. Razeto. Robust iris location in close-up images of the eye. *Pattern Anal. Appl.*, 8(3):247–255, 2005.

[109] G. Tsechpenakis, J. Wang, B. Mayer, and D. N. Metaxas. Coupling crfs and deformable models for 3d medical image segmentation. In *ICCV*, pages 1–8, 2007.

[110] K. Tsubota. Tear dynamics and dry eye. *Progress in Retinal and Eye Research*, 17(4):565–596, 1998.

[111] R. Tutt, A. Bradley, C. Begley, and L. N. Thibos. Optical and visual impact of tear break-up in human eyes. *Invest Ophthalmol. Vis. Sci.*, 41:4117–4123, 2000.

[112] M. E. Tyler, P. J. Saine, and T. J. Benett. *Practical Retinal Photography and Digital Imaging Techniques*. Butterworth Heinemann, 2003.

[113] A. Uchida, M. Uchino, E. Goto, E. Hosaka, Y. Kasuya, K. Fukagawa, M. Dogru, Y. Ogawa, and K. Tsubota. Non-invasive interference tear meniscometry in dry eye patients with sjogren syndrome. *Am J Ophthalmol*, 144(2):232–237, 2007.

[114] O. Veksler. Graph cut based optimization for mrfs with truncated convex priors. *CVPR*, 2007.

[115] T. Walter and J. C. Klein. Automatic detection of microaneurysms in color fundus images of the human retina by means of the bounding box closing. In *ISMDA '02: Proceedings of the Third International Symposium on Medical Data Analysis*, pages 210–220, London, UK, 2002. Springer-Verlag.

[116] J. Wang, J. Aquavella, Y. Zhao, and S. Chung. Tear dynamics measured with real-time optical coherence tomography. *J. Vis.*, 4(11):89–89, 11 2004.

[117] L. Wang, A. Bhalerao, and R. Wilson. Analysis of retinal vasculature using a multiresultion hermite model. *IEEE Trans. on Med. Imag.*, 26(2):137–152, 2007.

[118] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, 1995.

[119] M. Wels, G. Carneiro, A. Aplas, M. Huber, J. Hornegger, and D. Comaniciu. A discriminative model-constrained graph cuts approach to fully automated pediatric brain tumor segmentation in 3-d mri. In *MICCAI*, pages 67–75, 2008.

[120] R. P. Wildes. Iris recognition: An emerging biometric technology. *Proc. of the IEEE*, 85(9):1348–1363, 1997.

[121] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. *Computer Vision and Pattern Recognition*, 1:37–44, 2006.

[122] Wisedude. Anatamoy, structure and functions of the eyelid, conjuncitva, bulbar conjunictiva. *http://www.wisedude.com/health_medicine/eyelids.htm*.

[123] J. Xiao and M. Shah. Motion layer extraction in the presence of occlusion using graph cut. *CVPR*, pages 972–979, 2004.

[124] N. Yokoi, A. J. Bron, J. M. Tiffany, K. Maruyama, A. Komuro, and S. Kinoshita. Relationship between tear volume and tear meniscus curvature. *Arch Ophthalmol.*, 122:1265–1269, 2004.

[125] F. Zana and J. C. Klein. A multimodal registration algorithm of eye fundus images usingvessels detection and hough transform. *IEEE Trans. on Med. Imag*, 18:419–428, 1999.

[126] J. Zhou, W. Bischof, and T. Caelli. Road tracking in aerial images based on human-computer interaction and bayesian filtering. *SPRS journal of photogrammetry and remote sensing*, 61(2):108–124, 2006.

Index